昭和58年6月7日国鉄首都特別扱承認雑誌第6895号　昭和59年2月1日発行(毎偶数月1日発行)第2巻第1号

FUJITSU PERSONAL COMPUTER MAGAZINE
パソコン情報誌

Oh!FM

隔月刊　MICRO 7 8 11　'84年2・3月号
Vol. 7　480円

●特集●ディスク活用術

くれぬなら作ってしまえ！ アームロボ
マニピュレータの製作

蓉子の住所録

続・BIOSの使い方

新連載/マシン語ゲームの作り方

信頼と創造の富士通

友だちづき合いできるパソコン《FM-X》

ニュースです。話題です。富士通から、話題のMSX対応のパソコンが出ました。名前は《FM-X》。ゲームに、家事に、学習に、家中で楽しめる市販のソフトがいっぱい。MSX-BASIC搭載。16色カラーグラフィック表示。8オクターブ3重和音のサウンド機能。カートリッジスロットを装備etc。家庭用TVや、お手持ちのスピーカーと接続して、だれもが簡単に、すぐ楽しめます。いわば親しい友だちづきあいできるパソコン《FM-X》。ここから楽しいパソコンの世界がひろがります。

《FM-7》とつなげば、なお楽しい。

《FM-X》は、《FM-7》とつなぐことにより、ユーザエリアが32KBに拡張。《FM-7》側のプリンタが使用できます。また、《FM-7》の拡張BASICにより、MSXだけでは味わえないスピード感あふれるスプライト機能が実現。8オクターブ3重和音の立体感あふれるステレオサウンドも楽しめます。

新発売

友達みたいなパソコン
FM-X
エックス
¥49,800 (本体価格)

FM-7で、FM-Xのモニターになろう。
(人気No.1) (話題独占)

さらに盛りあがるぞハードとソフト。

発売以来、大人気、大好評の富士通の興奮パソコン《FM-7》。人気・実力No.1をうらづける、豊富な機能のハードとますます充実するソフト。価格も手頃。三拍子そろって、全国の青少年を、ビジネスマンを、おおいに盛りあがらせています。

締切り迫る1/31まで

富士通

君も《FM-X》のモニターになろう。

パソコンファン感涙のお知らせです。人気・実力No.1の《FM-7》を、今、お求めになると、富士通のパソコン《FM-X》のモニターになっていただけます。抽選により200名様、なんと期間は1年間。このビッグチャンスを、お見逃しなく。

●応募要項　《FM-7》の本体に添付されているアンケートハガキに必要事項と、その裏面下部に ①《FM-X》モニター応募 ②本体製造番号(=S/N 本体裏面に表示されています)をご記入のうえ、昭和59年1月31日(当日消印有効)までにお送りください。
＊発表は当選者へのご通知をもってかえさせていただきます。

富士通株式会社 半導体統轄営業部 〒105 東京都港区虎ノ門2-3-13 ☎(03)502-0161

人気No.1の興奮パソコン
FM-7 セブン
¥126,000
(本体価格・簡易言語ソフト付)

MSX
FM-X エックス

信頼と創造の富士通

興奮のハード。

パワフルな機能、使えるソフトで人気集中。

FMシリーズの高級機種《FM-11》は、オフコンにせまる本格派16ビットパソコンとして、EDP部門から現場まで幅広いビジネスユースに対応。パワフルな機能と使い勝手の良さで、先端派ビジネスマンに、エキサイティングな話題をなげかけています。

日本語処理機能が強化

●《FM-11》EXはOSとしてCP/M-86®を標準装備。漢字サポートを強化したF-BASIC86で日本語処理が容易になりました。また、漢字LEVEL II COBOL™も新発売。(漢字ROMカードはオプション)

●また、今評判の16ビットOS、MS-DOS(¥43,000)、136桁24ピン漢字プリンタ(¥315,000)、16ピン熱転写プリンタ(¥79,800)も新発売。

多彩なソフトで機能拡大

●TTY・BSC通信サポートにより、ホストコンピュータとのオンライン処理が可能です。●簡易言語ソフトFM CALCを標準装備。

●メインメモリ128KBを標準実装し、最大1MBまで拡張可能。

※CP/M-86®はディジタルリサーチ社、漢字LEVEL II COBOL™はマイクロフォーカス社、MS-DOSはマイクロソフト社の登録商標です。

ニーズにあわせてお求めやすい3タイプ

	EX	AD	ST
メインCPU	MBL8088-2 MBL68B09E	MBL68B09E	MBL68B09E
ミニフロッピィ	1ドライブ	1ドライブ	オプション
システムソフト	CP/M-86 F-BASIC	F-BASIC	F-BASIC (ROM版)
簡易言語ソフト	FM CALC	FM CALC	———

ビジネス用途を大きく拡げる高級パソコン。

FM-11

イレブン

EX ¥398,000(本体価格・簡易言語ソフト付)　AD ¥338,000(本体価格・簡易言語ソフト付)　ST ¥268,000(本体価格)

先端技術が夢中にさせる興奮パソコン。

FM-7

セブン

¥126,000(本体価格 簡易言語ソフト付)

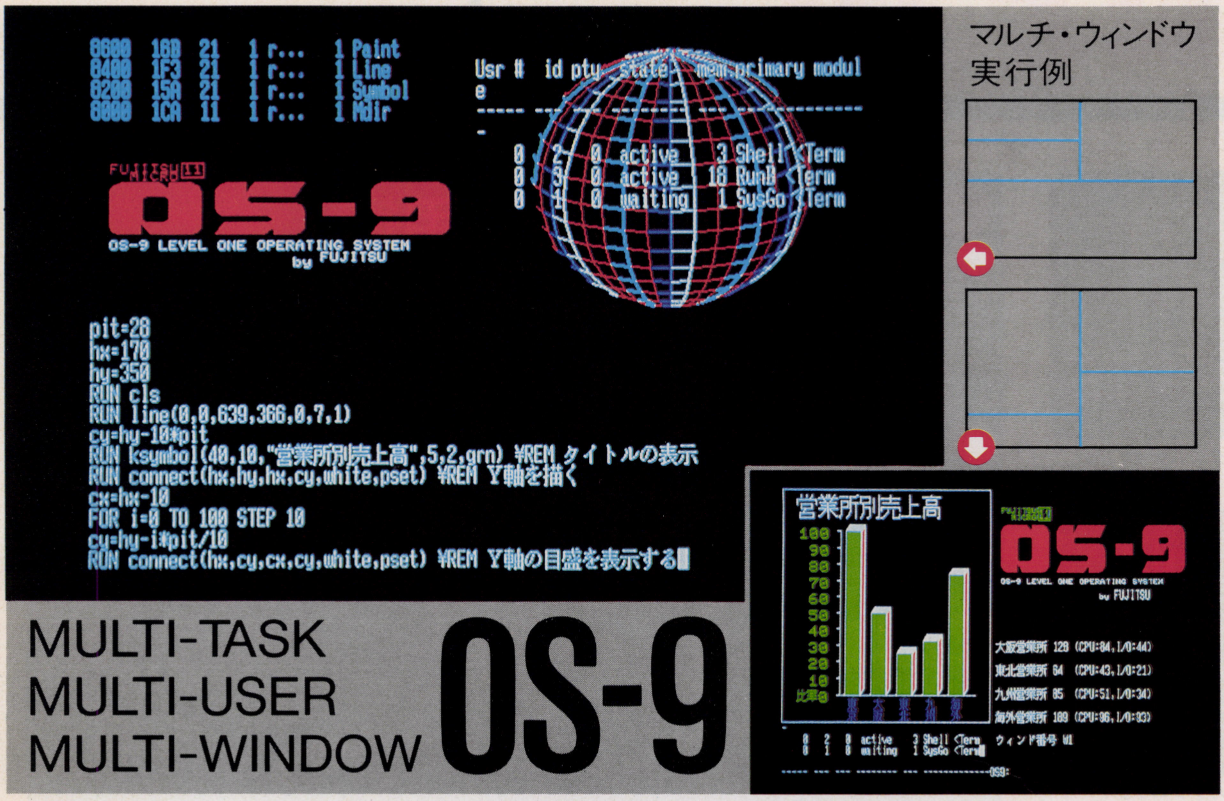
高級ホビーからビジネスまでの多才パソコン。

FM-8

エイト

¥218,000(本体価格)

感激のソフト。富士通のパソコン FMシリーズ。



こえてるビジネス、こえてるソフト。

一度に複数の作業を可能にする、オペレーティングシステム、OS-9。現在のミニコンの主流となっているUNIXの数々の特長を受け継ぎながら、マルチ・ウィンドウ、マルチ・タスク、マルチ・ユーザー（TSS）等、FMシリーズの機能をさらにパワーアップ。いままでのパソコンの概念をこえたビジネス機能を実現しました。

驚異のマルチ処理機能

一つの画面を複数に分割（マルチ・ウィンドウ）、それぞれの独立した画面で、データの検索、入力、出力、通信などの複数の処理を同時に実行することができます（マルチ・タスク）。しかも実行中のプログラム間でのデータの相互交換（パイプライン）など、処理効率は飛躍的にアップしました。また、OS-9は、FMシリーズの一台をホストとし、複数のパソコンを端末装置として同時に使用することが可能です（マルチユーザー）。

便利な漢字処理機能

ビジネスでの使いやすさを考えると、やはり漢字処理機能は欠かせません。OS-9は、BASIC09や、PASCAL等のプログラムリストで漢字表示ができますので、一層実用的なプログラム作成が可能です。

大容量記憶装置、ハードディスクに対応

膨大なデータ処理に威力を発揮するハードディスクユニットをサポート。LevelⅡにおいては、FM-11で最大1Mバイトまでのメモリ空間の利用が可能です。

※OS-9はマイクロウェア社、UNIXはベル研究所の登録商標です。

OS-9			
Level I	FM-7	(5D)	￥48,000
	FM-8	(5D、8D)	
	FM-11	(5D、8D)	
LevelⅡ	FM-11	(5D、8D)	￥58,000

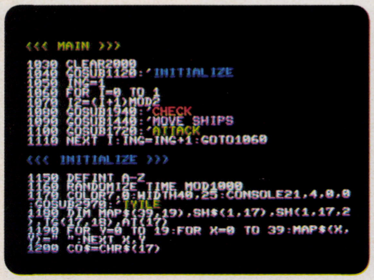
富士通株式会社　半導体統轄営業部　〒105　東京都港区虎ノ門2-3-13　☎(03)502-0161

●札幌営業所(011)271-4311 ●東北営業所(0222)64-2131 ●金沢営業所(0762)63-7621 ●長野営業所(0262)26-8222 ●静岡営業所(0542)54-9131
●名古屋営業所(052)201-8611 ●大阪営業所(06)344-1101 ●広島営業所(082)221-2288 ●九州営業所(092)411-6311

マイコンスカイラブ——FMシリーズのハードからソフトまで一挙に展示実演。あなたのパソコンのコンサルタントとしてご活用ください。
●東京・虎ノ門(03)591-1091　591-2561 ●東京・秋葉原(03)251-1448　251-1449 ●札幌〈時計台ビル〉(011)222-5476 ●札幌〈丸井今井〉(011)241-4185
●仙台(0222)66-8711 ●名古屋(052)221-6016 ●大阪(06)344-7628　341-0486 ●広島(082)247-3949 ●福岡〈開設準備室〉(092)471-7203

富士通

Oh!FM 目次



REM文おもしろテクニック（P. 103）

REM文おもしろテクニック
おまけゲーム（P. 107）

マシン語ゲームの作り方（P. 97）

特集 ディスク活用術

ディスク使用のメリット
——正しい見栄ディスクユーザーの在り方——木下淳博●17

FMシリーズにつながる
フロッピディスクカタログ——●21

ディスク用語集——●24

ディスクユーティリティ集

ディスク修復などに何かと便利な
ディスクエディタ「ZAPFM」——平野 剛●27

FLEXで開発している人に
FLEX→F-BASICファイルコンバータ——Simple Logic●31

知っていると差が出るディスクの知識がつく
スキューをかけるプログラム——鶴岡哲明●36

薄型FDUのノコギリ音でお悩みの方へ
ステップレート変更プログラム——Q-Laboratory●42

カタログ表示＆ディスケット再利用プログラム——山岸英人●44

ディスクおもしろプログラム——●20

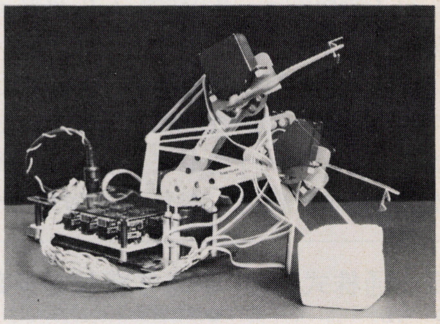
F-BASICに裏RAMコマンドを拡張
オールRAMモード活用法——是方研二●46

プログラムを見やすく——おまけゲームリスト付
REM文おもしろテクニック——佐藤進一●103

くれぬなら作ってしまえ！ アームロボ
マニピュレータの製作——米村英明●52

サンプルプログラムリスト付
ゲームプロセッサGP-7試用記——矢城貴志●72

6809機械語の書き方〔前〕——草津酔床●111


マニピュレータの製作（P. 52）

Vol.7　1984.2・3月号

今号の表紙
「YOU ARE MY VALENTINE」

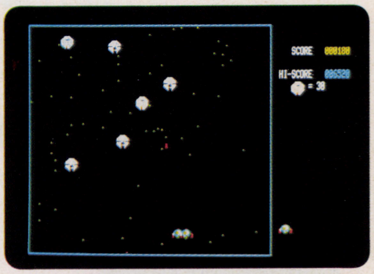
新連載
マシン語ゲームの作り方
笠作貴弥・小牧徳夫 ● 97

初歩の機械語入門
続・BIOSの使い方〔1〕 ―― 古田　昌 ● 88

F-BASIC中級入門
蓉子の住所録 ―― 菊池蓉子 ● 65

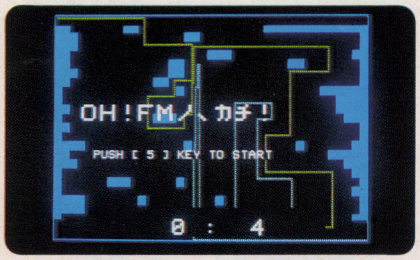
構造化プログラミングの理論 ―― 滝本康夫 ● 81

ショートプログラム集 ―― ● 123



ゲームプロセッサGP-7サンプルゲーム（P.72）

FILES強化プログラム（P.125）



生き残りゲーム（P.123）

FM-X 続報 ―― 本GO 誠司 ● 93
SOFT REVIEW ―― ● 116
(VISUAL ANGLE) ―― ● 14
マニュアルバグ情報 ―― ● 51
Q & A ―― ● 128
READER'S AREA ―― ● 130

Vol. 1～7 INDEX ―― ● 134
虫つくろい（前号の訂正） ―― ● 51
プレゼント情報 ―― ● 129
広告索引 ―― ● 160
EDITOR'S ROOM ―― ● 160

・CP/M, CP/M-86, MP/M は Digital Research 社
・MS-DOS は Microsoft 社
・FLEX は TSC 社
・UNIX は Bell Laboratories
・UCSD p-System はカルフォルニア大学理事会
・OS-9 は MICROWARE 社
・SB-80, SB-86 は LIFEBOAT ASSOCIATES
・WORDSTAR, MAILMERGE, SPELLSTAR, WORDMASTER, CALCSTAR, DATASTAR, SUPERSORT, INFOSTAR は Micro Pro 社の各メーカーの登録商標です。
・MSX マークはマイクロソフト社の商標です。

その他プログラム名，システム名，CPU名は一般に各開発メーカーの登録商標です。

本文中では"TM","Ⓡ"マークは明記していません。

本誌に掲載されたすべてのプログラムは，個人で使用する以外著作権法上，無断複製を禁じられています。

いま話題を呼んでいる！

ByteShopオリジナル A.D.A. soft

こんな価格でいいの？驚異の低価格

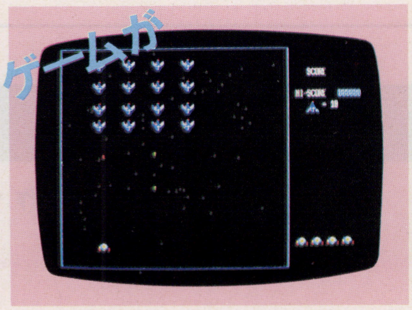
これがバイトショップの精神です。

ゲームプロセッサー **GP-7**

FM-7

君だけのオリジナル・ゲームが簡単にできる！

GP-7はパターンエディターと簡易型コンパイラーを備えた「ゲームソフトを作るため」のプログラムです。BASICのテキスト上に簡単な命令を書くだけで、オールマシン語（サブCPU直接制御）のスクロールゲーム（インベーダーやギャラクシアン類）を作ることができ、完成したプログラムは独立したものとしてファイルに保存できます。（アプリケーションゲーム2本付き）

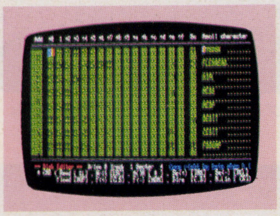


テープ版 ¥4,200
Disk版 ¥5,200

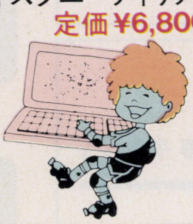
新製品

だいじょうぶマイフレンド

FM-7・8・11、PC8001（MK II）・PC-8801



ディスクユーティリティ
定価 ¥6,800



1枚のディスケットでFM・PCシリーズが使える！

FMシリーズの各プログラム
- FMシリーズのシステムDISKの作成及びVolcopy Program とSystem copyプログラム
- FLEX Binary Fileを読むプログラムとFLEXのディスケットのSector Dumpを行うプログラム
- FM-7の裏RAMをDiskと同じように使うプログラム
- FM-7, 8, 11用のメモリーエディター
- FM-7, 8, 11用のディスクエディター
- FM-7, 8, 11用のID表示付Volcopyプログラム
- FM-7, 8, 11用Disk Directory Dump
- 不用なプログラム及びファイルをKillするプログラム
- FM⇄FM, PC⇄PC, FM⇄PCのファイルコンバーター

PCシリーズの各プログラム
- PC-8001 & PC-8031-2W(80S31)用DISKエディター及びPC-8031内のメモリーエディターです。トラックリード，トラックライト付でPC-8031内の指定アドレスに実行を移すことができます
- メモリーエディター
- File Dump
- Program & File COPY
- プログラムとファイルの使用クラスターをDump
- DISKのFormat & BACKUP
- DiskのFATの使用状態をDump
- Fileの属性のSet, Reset, Killを行う

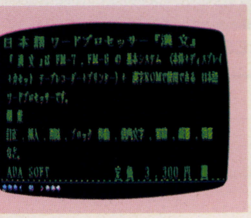
※PC-8801もしくはPC-8001mk2はN-BASICモードで使用できる。

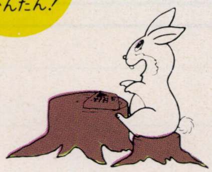
日本語ワード プロセッサ **"簡文"** テープ版

FM-7・8

"簡文" かんたん！かんたん！

定価 ¥3,300





ディスクを持っている人も買う。これほんと？

特長

- FM-7,8の基本システム（本体漢ROM付・ディスプレイ・カセットレコーダー・シリアルドットプリンター）のみで使用することができる。
- INS, DELなどのキーがBASICのエディター同様の機能を持つ。
- 漢字の呼び出しは音訓いずれからでも可能、語を品詞別に入力することによりよりスムーズに目的の漢字を捜せる。
- シリアルドットプリンターで漢字プリンターと同品質の印字が得られる。
- すべてマシン語でプログラムを組んであり高速処理ができる。

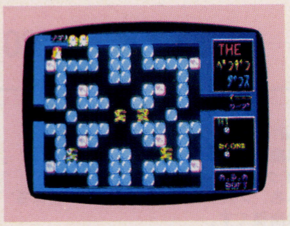
システム仕様

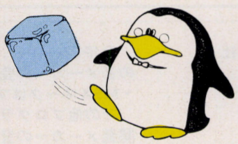
●**入力方式**／かな漢字変換（文字単位、品詞別）動詞については活用語尾とともに入力可能●**使用文字数**／漢字2,965種（JIS第一水準）、その他463種●**文章作成機能**／訂正、挿入（INSキー）、削除（BS、DELキー）、ブロック移動、１行挿入／削除、改行（ESCキー）、スクロールＵＰ／DOWN、タブ（TABキー）、倍角文字、罫線処理●**印字機能**／縦書／横書、行間変更、倍長文字●**ディスプレイ**／40字×８行、●**文章記憶力**／48ページ（400字詰原稿用紙 約38枚）カセットテープに記録可

ザ・ペンギンダンス

FM-7・8

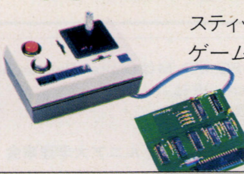
定価 ¥3,300





南極に現われた"火を吐く"ファイアーモンスターを君の指示により氷塊を蹴とばして退治する。「ペンちゃん」頑張れ！
ジョイスティック（ADA-7）、テンキーともに対応

JOYSTICK（ADA−7）



スティック＋アナログボード＋ゲームソフト（ペンギンダンス）
ADA-7002とジョイスティックの組合せです。
X-Y2方向、スイッチ2個 ¥9,800

ByteShopで広がるコミュニケーション

プリンタ、プロッターやフロッピーの周辺装置から、デジタイザ、漢字タブレットその他のインターフェイス相談。O・S-9、Flex、CP/M、アプリケーションソフトまでなんでもOK。

技術者募集

8bit,16bit, ハード，ソフトウェア技術経験者
"あなたの能力を十分に生かせる"
TEL 03-257-6309

A.D.A.オフィス

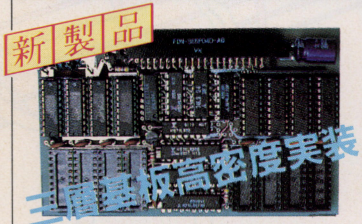
〒101 東京都千代田区神田須田町2―23―1
芝崎ビル５F（バイトショップ・サポート内）

通販
によるお求めは送料¥500を入れ現金書留にてどうぞ。
☎03-257-6337

マイコン・パソコン専門店

FMシリーズのシステムアップは Byte Shop

C-MOS RAMボード(ADA-7006)



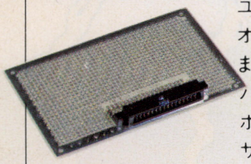
- I/Oアドレスマップド接続の C-MOS RAM ボード です。
- BASICプログラム、 データ等を記憶させる外部記憶エリアとして使用可能です。
- 32KBの容量で、 バッテリーにて停電保証しています。(16KB実装)
- 保証時間：200時間(32KB実装時)
データの入出力は、BASIC又はマシン語の命令による。(サンプル プログラム付)

PIA学習セット

ADA-7005
＋
ADA-7003
＋
接続ケーブル

特価 ¥32000

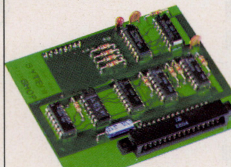
汎用ユニバーサルボード ADA-7001



ユーザが自由な回路をFM-7オプションスロット内に組込めます。TTLIC20コ相当のユニバーサルボードです。ガラスエポキシ、スルーホール、 基板サイズ 125×77、32P・40P共用可能、2.54ミリピッチ。

¥3,200

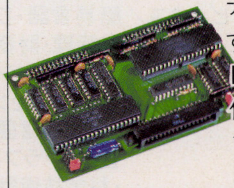
アナログボード ADA-7002



8ビット、4チャンネル、ANPORT命令サポート、低価格A/Dコンバータボードです。

¥4,800

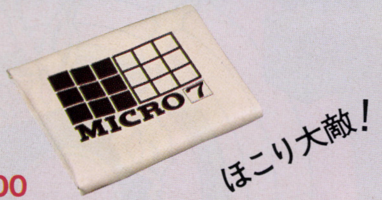
PIAボード ADA-7003



デジタル信号を入出力するためのボードです。PIA(MB-8874相当)2個実装、割り込み可能、1ポートのみバッファ付き。

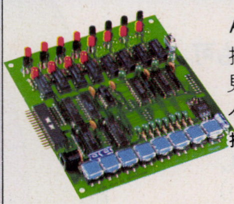
¥12,800

高級ダストカバー FM-7用



¥1,200

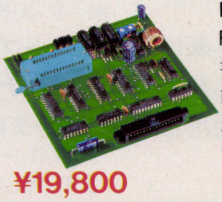
接点入出力ボード ADA-7005



ADA-7003(PIAボード)と接続することにより、目で見るペリフェラル学習や種々なコントロールができる。
接続ケーブル (¥2,500)

¥19,800

ROMライターボード ADA-7004

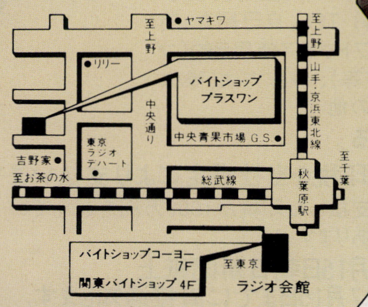


EPROM書込み器です。FM-7オプションスロットに挿入するだけで、即使用可能です。 専用ソフトウェアが付属しています。(カセットテープ)対応EPROM：2716/2732/2764

¥19,800

関東バイトショップ

秋葉原なら Byte Shop



(営業時間 A.M. 10:00～P.M. 7:00)

年中無休

バイトショップ コーヨー

バイトショップ プラスワン

クレジット、即決!!

大特価！
超低金利！

お好きな月に1回
さらに3～60回の長期分割可
即決クレジットでその場でお持ち帰りになれます。
(クレジット希望の方は印カン・身分証明書等お持ち下さい)

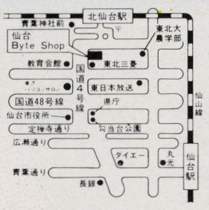
- 各種クレジット取扱中。
- お気軽にご利用ください。

DC　UC　VISA

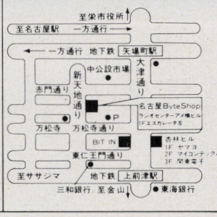
日本全国バイトショップ

●仙台バイトショップ	宮城県仙台市提通雨宮町3-18 〒980	☎0222-33-0256
●伊勢崎バイトショップ	群馬県伊勢崎市今井町755 〒372	☎0270-23-2302
●岡谷バイトショップ	長野県岡谷市幸町6-11(五十川ビル)	☎02662-3-1075
●バイトショップ プラスワン	東京都千代田区外神田1-8-11	☎03-257-6337
●関東バイトショップ	東京都千代田区外神田1-15-16(秋葉原ラジオ会館4F)	☎03-257-6341
●バイトショップ コーヨー	東京都千代田区外神田1-15-16(秋葉原ラジオ会館7F)	☎03-255-6504
●名古屋バイトショップ	愛知県名古屋市中区大須3-30-86(ラジオセンターアメ横共同ビル)	☎052-263-1629
●大阪バイトショップ	大阪市浪区日本橋3-3-5(カトウビル4F)	☎06-644-6655
●福岡バイトショップ	福岡県福岡市博多区博多駅前2-13-23(扇寿ビル)	☎092-474-5778

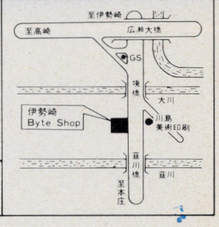
名古屋



仙台



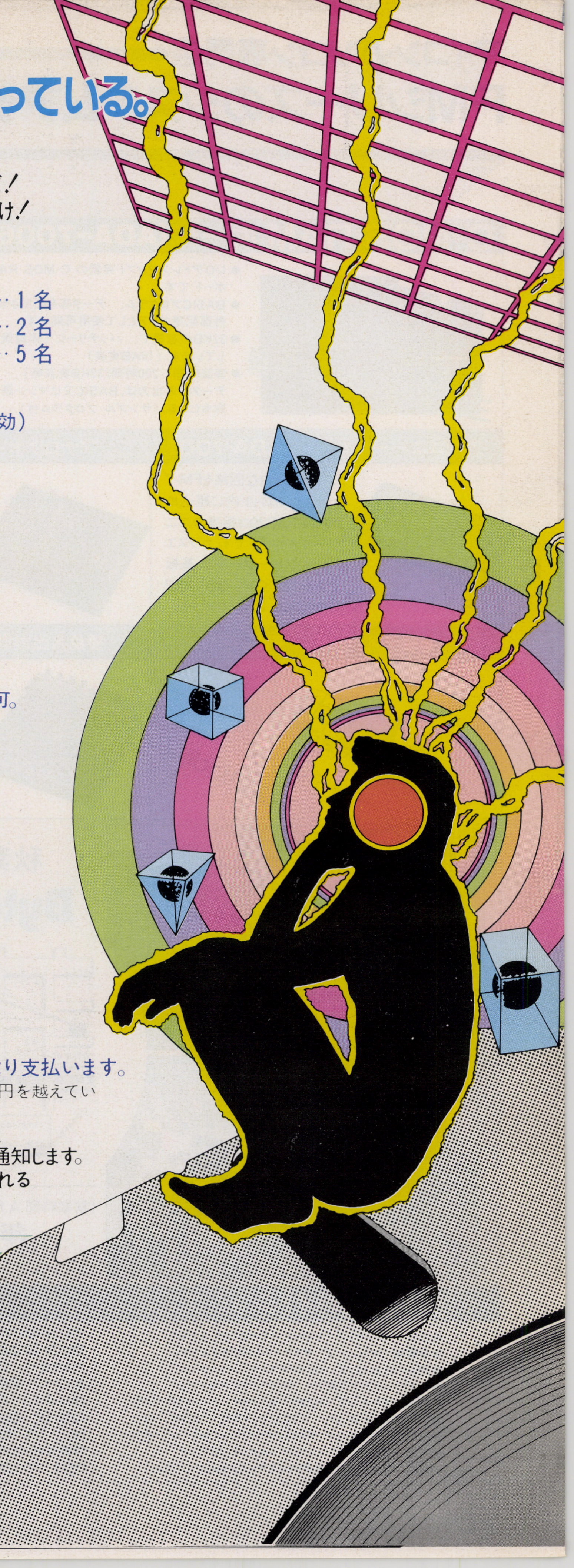
伊勢崎



第1回 パソコンソフトウェア技術コンテスト

パソコンの未来は ソフトウェア技術者が握っている。

独創的なアイディアを持っているあなた。
優秀な技術力を持っているあなた。
今、あなたのアイディアと技術力を発揮する時だ！
時代の流れを変え、新しいパソコンの世界を開け！
名声と富があなたを待っている。



〔賞金〕

■最優秀ソフトウェア技術賞100万円………1名
■優秀ソフトウェア技術賞50万円……………2名
■ソフトウェア技術賞20万円…………………5名

〔募集要項〕

■募集期限…昭和59年4月末日(当日消印有効)
■テーマ……自由

例) ビジネス／簡易言語・ワープロ・データベース・顧客管理・財務管理・販売在庫・業種別アプリケーション・その他
ユーティリティ／アッセンブラー・コンパイラー・プログラム言語・OS・開発ツール・ファイル管理・グラフィックエディター・通信制御・科学技術計算・その他
ゲーム／ロールプレイング・アドベンチャー・シミュレーション・シューティング・チェイス・ボード・スポーツ・ギャンブル・その他

■応募は個人、グループ、団体、法人等いずれでも可。
■応募作品はオリジナルに限ります。
■使用機種…8ビット又は16ビットCPUでパソコンと呼ばれ、一般に流通しているものとします。但し特別な(市販されていない)装置を増設しないこと。
■応募方法…下記の事項を明記した書類とプログラムが入ったディスケット又はテープを添えて、コンテスト事務局までお送り下さい。
(記)氏名、性別、年令、住所、TEL、勤務先(学校名)・TEL、自己紹介、使用機器構成、ソフトウェア使用目的(プログラム仕様書等)、操作方法(操作説明書等)、その他
■著作権は作者に帰属する。
■商品化に際して作者には印税を当社規定により支払います。
〈現在、当社で支払った印税が1000万円を越えている作品が数点あります。〉
■発表………●昭和59年7月15日朝日新聞紙上
●入賞者には7月15日までに直接通知します。
●昭和59年8月15日以降に発売される主なマイコン雑誌誌上。

〈主催〉
株式会社エニックス

お問い合せ・送付先
株式会社エニックス技術コンテスト事務局
〒160 東京都新宿区西新宿7-1-8
☎03(366)4251(代)

ハミングバードアドベンチャーゲームシリーズ

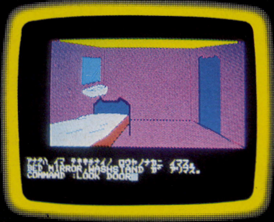
ABYSS アビス

新発売

Humming Bird Adventure #3

(富士通FM-8・7用5インチディスク版)

標準宇宙紀3001年。
銀河系外宇宙
〈9惑星連合〉の運命を
賭けて、
1人の戦士が、
深宇宙へと旅立った。



深宇宙探査員のあなたは、
最高機密指令〈ミッション:ABYSS〉
の任務遂行中、
宇宙犯罪シンジケートに捕えられ、
記憶を消されてしまう。
星の位置も、名前も、まったく不明。
早く母船に連絡し、
シンジケートの基地を
壊滅せねばならない———。
手がかりも武器もない、
たったひとりの脱出作戦が始まる。

¥9,800

There is no relief for you····

た、大変!フィアンセが海底人にさらわれちゃった。

ハミングバードアドベンチャー#1

'THE PALMS'' ザ・パームス

(富士通FM-8 & 7用・5インチディスク版) ¥11,000

手がかりは、ほとんどゼロ。謎の海底世界へあなたは単身潜入する。頼りはあなたの知力のみ。

旺文社パソコンゲームランキング 和製アドベンチャーゲーム 第3位

早く、早く!〝魔法の水〟を手に入れないと、姫が死んじゃう。

ハミングバードアドベンチャー#2

ザ・ナイト・オブ・ワンダーランド

''THE KNIGHT OF WONDERLAND''

(富士通FM-8・7用5インチディスク版、NEC PC-9801用5インチ両面 or 8インチディスク版) ¥11,000

あなたは、騎士。姫の病いをなおす"魔法の水"を探す旅へ。不思議に満ちたワンダーランドが、迷路になってあなたの行く手をはばむ。さて。

旺文社パソコンゲームランキング 和製アドベンチャーゲーム 第1位

関西のパソコン・キッズたちへ、

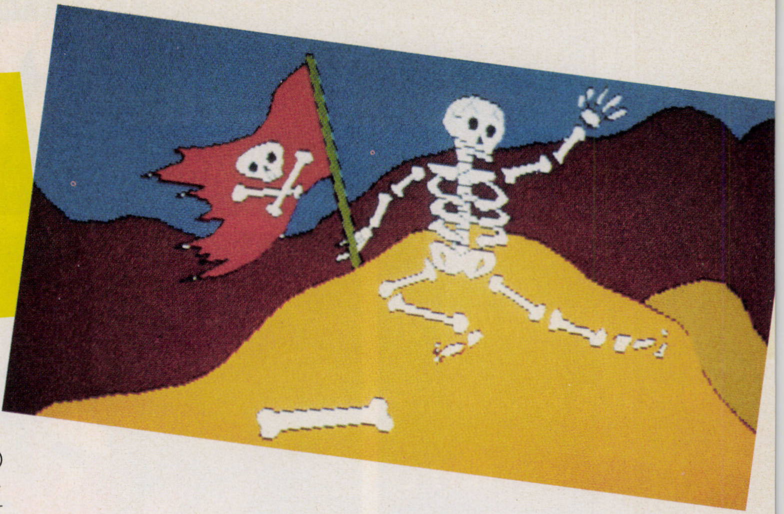
ショップも、よろしく。

ハミングバードアドベンチャーゲームのスタッフたちが待ってますよ。ブラッと、ぜひ一度。

(販売店募集中!)

Humming Bird Soft
(株)エム・エー・シー コンピュータ事業部
〒530 大阪市北区曽根崎2丁目2番15号 ☎06(315)0541

おもしろ体験記



君塚泰宏(都内在住、高校3年)
カラーの豊富さと、ペイントのスピードにただアゼン。全ての画面に意味があり、あらゆるキーワードが最後になってビシッとつながってくるストーリー展開の確かさもスゴイですね。

小久保秀次朗(都内在住、高校1年)
絵の鮮やかさと枚数もスゴイけど、にも増してキーワードの設定の巧みさに感心しました。いきなりグッグッと引き寄せ、最後まで緊張感を持たせるストーリーの確かさは、まるで推理小説のような迫力があります。あきさせません。

小久保　ただ入力の仕方が今までのアドベンチャーとはちょっと違う感じがする。英語入力でありながら、文法はまるで無視してるみたいな。

君塚　だから、さ‼変に英語力が無くとも楽しめるじゃない。ボクなんか独語専攻だから、単語だけで〝何をどうする〟式の会話型入力方式は助かります。間違うと、エラー・メッセージが出てくるし、引手に文法に気をつかわずにナンとなくフィーリングで入力して行けるものネ。

小久保　だから、いっそのこと日本語入力にすれば……。

竹部　それはネ、日本語にすると入力文字がどうしても多くなるから、むしろ中学生並みの単語知識で、ポツンポツンと入力して行けばいいように工夫したつもりなんだ。

君塚　う～ん、それにしても疲れた。もし竹部さんのサジェスチョンがなければ、ひと月やふた月は優にかかりそうな。下手をすると、パビリオンの中をあっちグルグル、こっちグルグル。

小久保　永遠に出られそうもない。そんなパビリオン5つもあって、そのひとつひとつのパビリオンが、従来のアドベンチャーゲームの1本分はある。そんなモノホンの迫力を感じましたね。

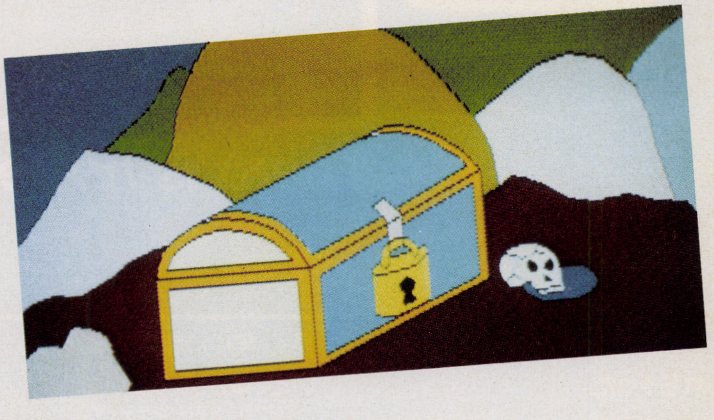
竹部　そこは、つまり制作者とユーザーの知恵比べというかね。それにハドソンとしても、今後のディスク時代に対応した揮心の第1作目ですから、力は入ります。デゼニランドはその意味で大サービスなんですよ。

君塚　ところで、この114枚の画面。これ全てに意味があるんですよね。最初の方は割と優しいヒントがもらえるからスイスイ行っちゃうけど、後になればヒントも少なくなるし、さてどうしようという時、それまでの画面やキーワードを覚えてないと立ち往生しちゃう。

小久保　ボクの場合は、途中で、あ、これはメモしておく必要があるなとネ。例えば、六つの店で買い物をする場面で、買い物リストを作るとか、全てのキーワードをメモっとくとか。そうしないと、これは解けない。

君塚　ウーン、いやはや何とも。全部で114画面の全てに後々必要になってくるキーワードが隠されているんだからね。それに、全部のキーワードが最後でビシッとつながってくるところに、ホンモノのアドベンチャーを感じます。ストーリーが生きてる‼

小久保　ウン、生きてる。いきなりグッグッと引き込み、最後まであきさせない。デキの良い推理小説並みの迫力です。ス・ゴ・イ‼



対応機種

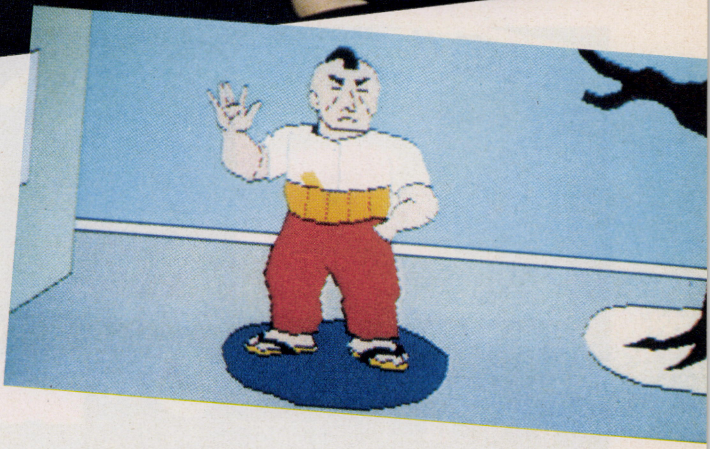
SHARP	X1-D	3″FD	¥5,800
	X1-C	T	¥4,800
富士通	FM-7	5″FD	¥6,800
		T	¥4,800
NEC	PC-9801	5″FD	¥6,800
	PC-8801	5″FD	¥6,800

HUDSON GROUP
HUDSON SOFT®
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

デゼニランド

気分は、デゼニランド
このアドベンチャーゲームは、ホンモノです!!

竹部隆司(ハドソン・ソフトエンジニア)
1週間や2週間でGAME OVERになるんだったら、アドベンチャーの意味がないんですよ。ストーリーと絵の面白さを充分楽しんでもらいたいですね。



竹部　ストーリー・アドベンチャーというのは、要するにキミが主役になって、途中幾多の困難に見舞われながらも、ある与えられた目的を達成するという所、基本的なストーリーを、マイコン用にコーティングしたもの。

で、普通この種のゲームにはオカルトハウス的なインドアものが多いけど、「デゼニランド(出銭ランド)」の場合は、言ってみればアウトドア・アドベンチャーです。途中海あり、川あり、空ありで、とにかく広がりのあるゲーム展開が売りもの。

舞台は突然ディズニーランド（千葉県）にワープして、このディズニーランドをパロった五つのパビリオンのどこかに隠された秘宝「三月磨臼(ミツキマウス)」を探し当てるべく、まずはデゼニランドに乗り込んだ二人の少年に、デゼニランド・ショックを語ってもらいます。因にオレ、制作者の竹部です。

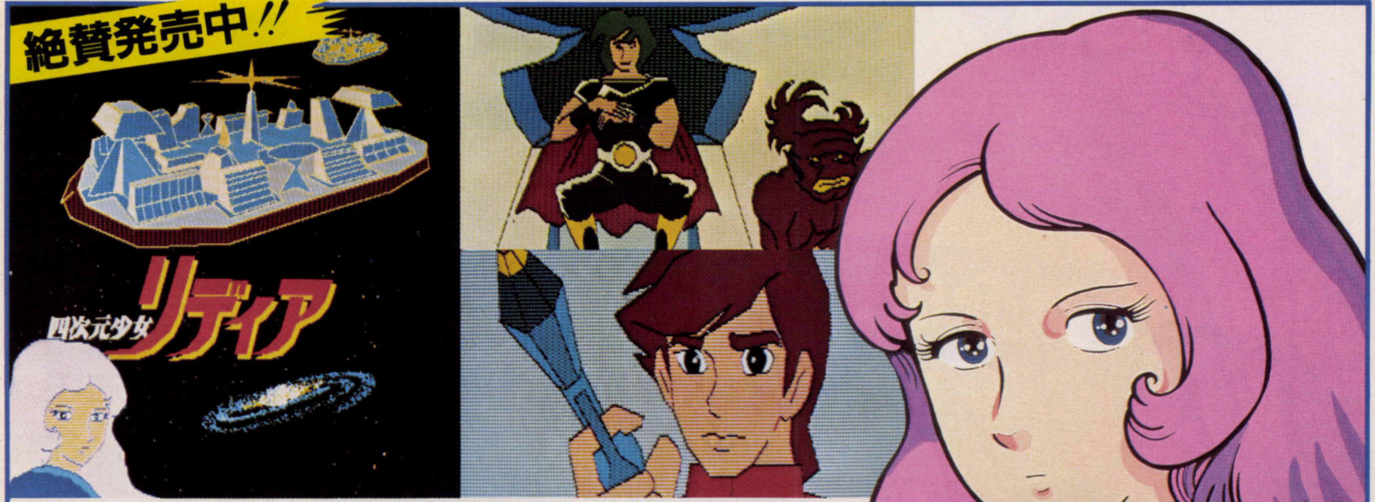
小久保　とにかく「ワー速い、きれい!!」がボク達の一致した感想ですね。それと、ナンとなく本物のディズニーランドへ行った気になる、というか、あ～行ってみたいとゆ～よ～な、逆に言うと実際に行ったことのある人がプレイすると楽しさ倍増といった感じですね。

竹部　実際に現場を取材しましたからね。というよりは、行ってみてものすごく面白かったから、それがこのゲームを作る直接的な刺激になりましたよ。

小久保　あ、やっぱり。実にリアリティがあるもんネ、ひとつひとつの場面に。ひと口に言うと、凝ってるって感じ。

君塚　リアリティとそれからペイントのスピード、色の豊富さ……中間色をこのくらい豊富に使って、しかもこれだけ速いというのはやはりディスク版だからかな。それにオールマシン語だから。同じディスク版でも、ボクの持ってるアップルのソフトより速いもの。これには驚いた。キーワードの入力が正しいと鮮やかな画面が次から次へ、パッパッパッだもんね。しかもそれが全部で114枚……!!

CHAMPION SOFT

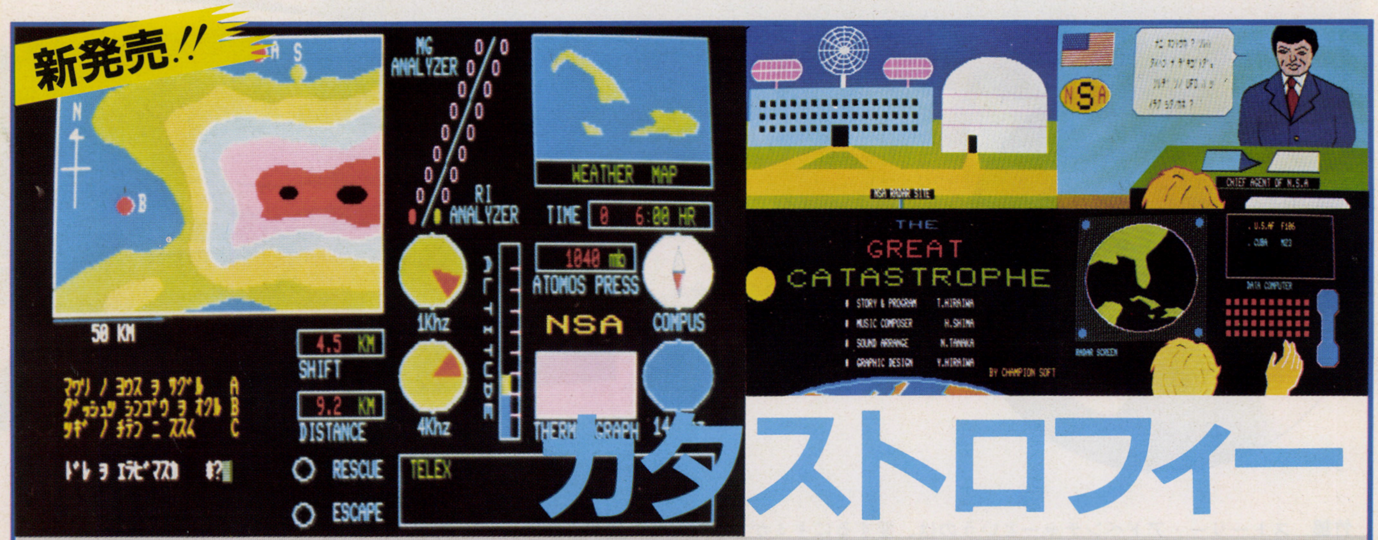


宇宙暦1001年。君はバミューダ・オブジェクトチームの隊員として、魔のトライアングル究明に乗り出した。そして、恐るべき現実を直視する！　アドベンチャー＋リアルタイムのマルチプレイに、4次元グラフィックを載せて、ついに。

立体シーンを含む50の画面。プログラムサイズ300K！

四次元少女リディア

PC-8801・mkII
FM-7〈各機種とも立体メガネ付〉
テープ版(4本組)4,800円
ディスク版……6,800円
X1D………6,800円



カタストロフィー

19XX年、アメリカ国家安全保障局レーダーに、UFOキャッチ！
調査に乗り出した君は、機体の残骸から、恐るべき事実を発見する。
100KBを越える分割プログラム。本格アドベンチャー登場！
■製作スタッフ／平岩徳一・島・田中 ほか

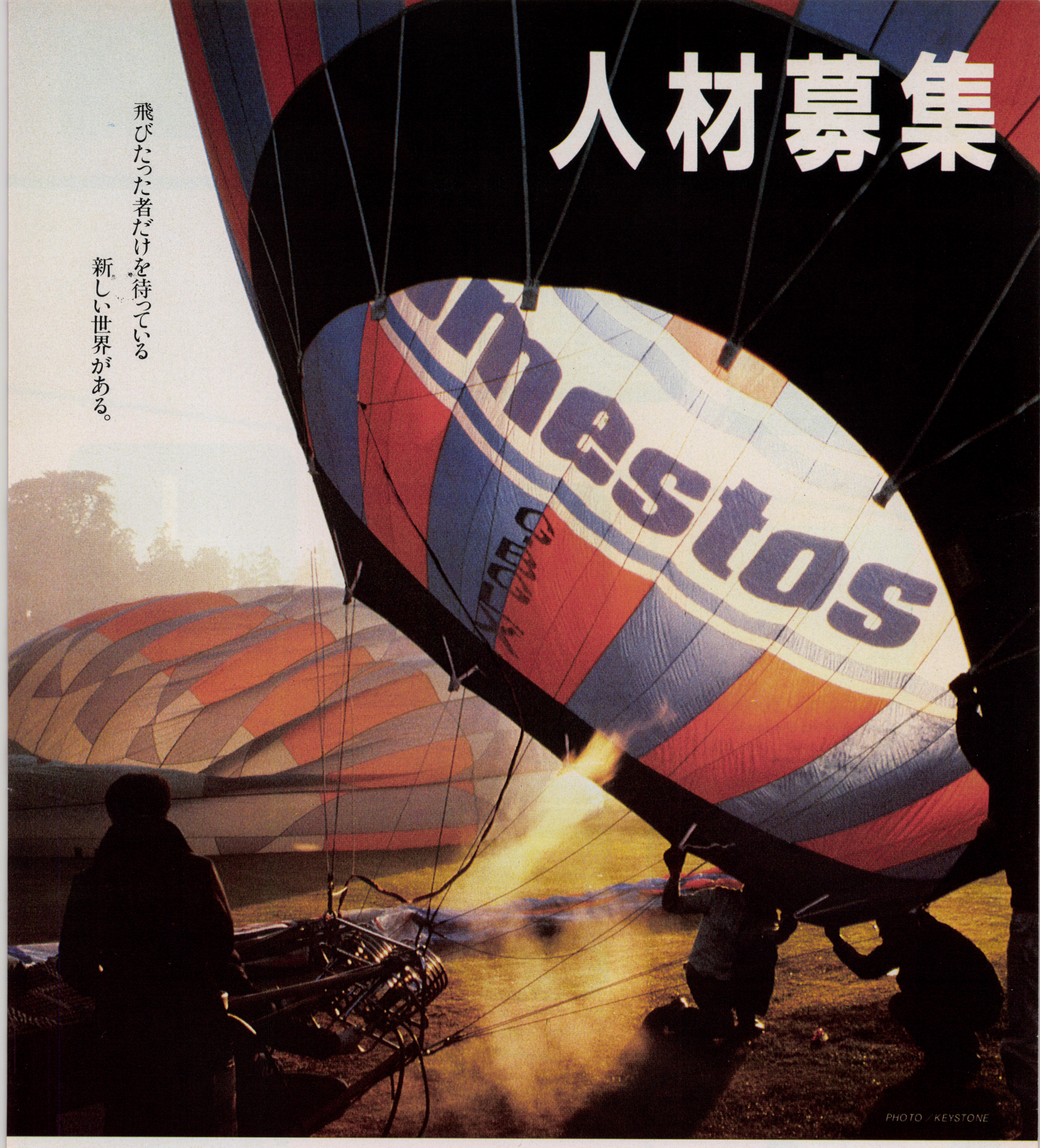
●FM-7　テープ版(3本組)5,800円
　ディスク版　7,800円
●PC-8801　テープ版(3本組)5,800円
　ディスク版　7,800円

販売代理店募集中

制作　株式会社チャンピオンソフト
〒530 大阪市北区西天満6丁目1番12号瑞穂ビル
TEL(06)365−9900　ファックス(06)361−1176

発売元　チャンピオンソフト販売株式会社
本社・通販部　〒657 神戸市灘区篠原北町1丁目5番13号
TEL(078)801-2727　ファックス(078)882-0555
東京営業所　〒111 東京都台東区蔵前4丁目33-7山愛ビル
TEL (03)866−8447　ファックス(03)851-4880

⊙お求めは、お近くのマイコンショップまたは、チャンピオンソフト販売㈱通販部宛に品名と機種、氏名、住所、電話番号を記入の上、現金書留にてお送りください。（送料は無料サービス）



マイクロコンピュータ技術者募集

日本ソフトバンクでは、出版事業の拡大により、マイコン技術者を次の要領で募集します。ハードについて自信のある方はどしどしご応募ください。

■募集人員／数　名　■待　遇／当社規定による

■応募方法　履歴書(写真貼付)を下記あてに郵送してください。面接日は追って通知します。なお、応募の秘密は厳守します。

■あ て 先　株式会社日本ソフトバンク人事室
〒102 東京都千代田区九段南2-3-14 TEL. 03(263)3665

SOFT BANK

株式会社 日本ソフトバンク

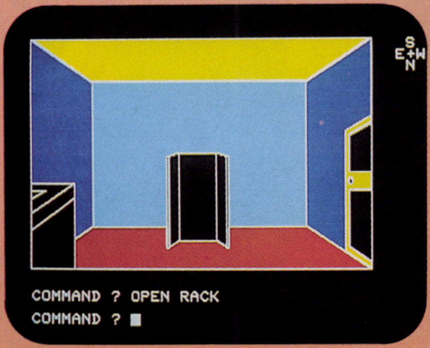
SOFT REVIEW VISUAL ANGLE

PIPE MAN

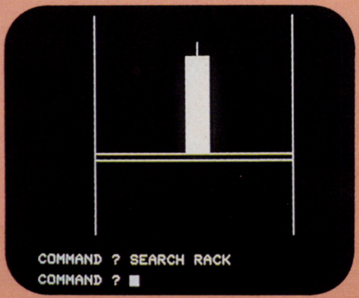
水もれを修理中，ヘラヘラがせまってくる！

はしごを通るとヘラヘラを食べられる。得点を増やすチャンスだ。なおこのソフトを5名の方にプレゼントします。詳細は129ページ。

ミステリーハウスI

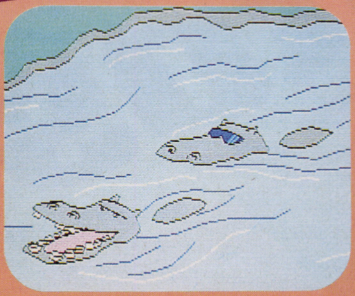


ミステリーハウスの中。ラックがある。開けてみよう。



ろうそくがあった。さてどうするか？

デゼニランド



ジャングルパビリオンの中。サングラスをかけたカバが登場。

お化け屋敷。ガイコツが挨拶してくる。話をしたら何かヒントをもらえるかも。

スペースモモタロウ。なんたって可愛いんだから。

EARTHBOUND

タイトル画面。なかなか見せる。ゲーム全体に猫が出てくる。

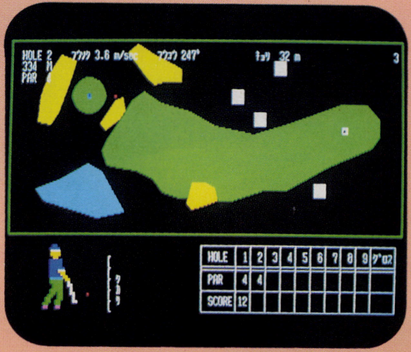
川の番人“CLOUDHANPH”。彼（？）を怒らせたらゲームエンド。気をつけて。

コスモクラッシュ

手前が自分の機。勝利めざしてがんばろう。

SOFT REVIEW(116ページ)に掲載したゲームソフトの中から，一部を写真で紹介する。価格，発売元などは本文を参照していただきたい。

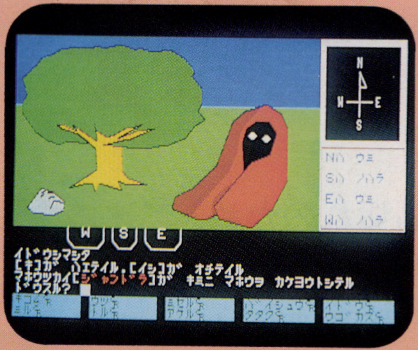
EXCITING GOLF (GAME PACK)



2ホール目の3打。グリーンに乗るかどうか（1ホールはお恥ずかしいスコアで……)。

グリーンに乗った！　さあパーできるか。思わず熱中ソフト。

不思議な旅



魔法使い“ジャンドラ”。ぐずぐずしてると猿にさせられる。

みゆきTHE勝負師

可愛いみゆきちゃんと勝負，勝負。あ，そこの人，よだれをたらさないでください。

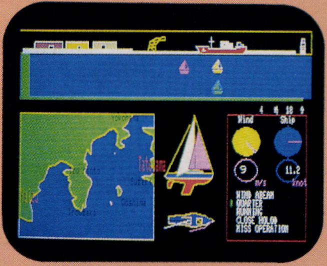
BEAN'S JACK

ジャックと豆の木を思い出して。でもノスタルジーにひたっているとカラスにやられます。

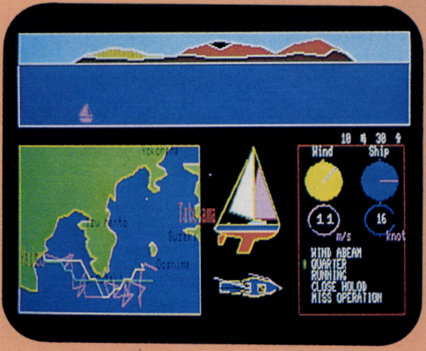
BLITZ KRIEG

慎重に作戦をたてないと，あっという間に自滅。

ヨットレース



焼津港をスタート。今のところ順調。



大島沖通過。左下に今までのコースが表示される。それにしてもひどい。

悪女かまきり

アダルトムード漂うタイトルデモ。未成年おことわり。

お待たせしました!

Oh!FM 別冊

FILES Oh!FM

㈱日本ソフトバンク発行

A4変型　340ページ
定価1,440円
雑誌コード　02200—11

Oh!FM Vol. 1～3品切れのため，ご迷惑をおかけいたしておりましたが，このたび，内容別に再編集しました。

主な内容

●Oh!新機種
FM-7のすべて
本格派パソコンFM-11の全容

●Oh!マシン語
6809マシン語入門
中間色PAINTもどきプログラム
サブシステムプログラム転送・実行
高速3Dグラフィックス
BIOS資料

●Oh!BASIC
初心者のためのF-BASIC
　プログラミングアプローチ
ソフトウェア設計 法入門講座
三次元パッケージのアルゴリズム

●Oh!ゲーム
モノポリー
ジミーラミー
川下り
B-1核爆撃機
アルデバランパート1

●Oh!DOS入門
OS-9+BASIC09
FLEX入門
CP/M-80とその使い方

●Oh!ソフト紹介
点検・ワープロソフト
グラフィックソフトパッケージ

●Oh!ハード
各社ドライブユニットの
　比較検討
FM-7ハードウェア
　回路図と解説

特集 ディスク活用術

ディスク使用のメリット

――正しい見栄ディスクユーザーの在り方

木下淳博

友人などにディスクを持っている人がいて，ディスクのメリットは充分わかっていながらも，先立つものがなくてしかたなくテープでがまんしている方，そういう人はこれを読む必要はありません。読まないでください。腹が立って，胃かいようになるだけです。

本稿は,「ディスク?　そんなものなくたってテープで充分だよ」と思っているあなたのためのものです。

まず最初に

言っておきたいのは，現在の状況からいって,

『ディスクがなければパソコンではない』

ということです。

テープだけでなんとかなることと言えば，体力と反射神経型のピコピコゲームや，電卓に毛が生えたくらいの機能しかない事務処理ソフト・計算ソフトなどしかないでしょう。大体のものは，もしディスクが使えたら，もっとすごいものになるのになー，というものばかりなんです。

その最大の原因は，4～5分間という，信じられないほど長い時間をかけてデータをテープに記録しないかぎりすべてのデータは数十Kという信じられないほど狭いメモリ上で処理しなくてはいけない，というところにあります。

そもそも，コンピュータのメリットは,

- 大量のデータを処理できる。
- スピードが速い。
- 複雑なことを繰り返しやるときに，人間の手間が省ける。

などでしょう。ディスクを使わないでテープを使うということは，そのうちの2つのメリットを最初から放棄していることになるんですよ。

ですから，今まで私が会った有能な人には,

「いやー，お金がなければテープでも仕方ありませんよ。」

などと言う人は，ほとんどいません。**ディスクを買うお金がなかったら，パソコンを買う必要がほとんどない**ということです。

ですから，不幸にして本体とディスプレイ，データレコーダしか持っていない人は,

「いや，これでジューブン。」

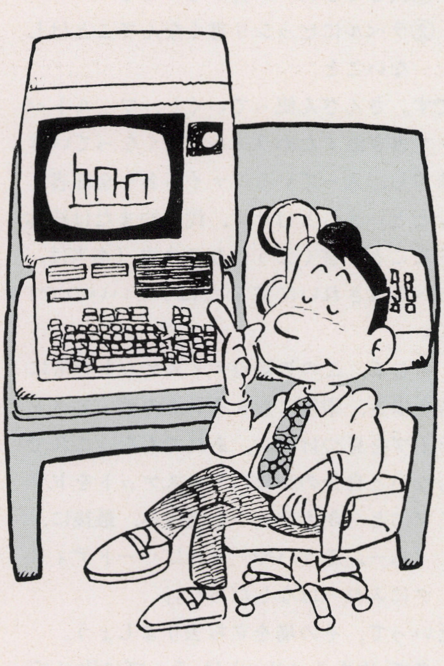
などと言わずに,

「いや，ちょっと，ヘッドのアジャストが思わしくなくてね。修理中なんだ。」

とか,

「あっそれ?　それねー，VAXの端末として使ってるんだ。」

と言って，見栄ディスクユーザーをやりましょう。ただし後者の場合，モデム等の通信装置が接続されていないとバレることがあります。



さて，それでは具体的にディスクのどこが良いかをみていきましょう。まずは,

スピードが速い

ということです。これは当たり前のことですが，やはりトップに書くべきことでしょう。

ロードやセーブが速いということは，市販ソフトを走らせるだけのときにも，タバコを吸うほど待たずにすむというメリットがありますが，自らプログラムを開発するときにこそ，威力を発揮するものなのです。

プログラムを作ったことのある人はよくわかると思うのですが，ある程度までプログラミングが進行すると,

「ここで停電が起きたらパァだなー」

という不安が頭をよぎるのです。これは精神衛生上，非常によくないことですね。そこで一段落ついたところでセーブするわけです。ところがこれがテープだと，数分間待たなければならないのです。しかも，テープの場合，修正前のプログラムが入っているテープの上に書き込むのは危険ですから，別のテープにセーブすることになるのです。テープがもったいないから，今までの次に入れたり，裏面に入れたりすることもできますが，そうすると，頭出しにまた時間がかかるでしょう。考えただけでも胃が痛くなってきます。

こんなわけですから，安全のために2本セーブするなんてことは，よっぽど心配症な人か，ヒマな人しかしませんね。これがディスクなら数秒から数十秒でできるのです。ファイル名の後にbをつけて，バックアップ用のファイルを余分にとっておくことくらい，わけありません。通常，マスタとバックアップの両方を作るようにしておけば，すべてを破壊してしまうことはな

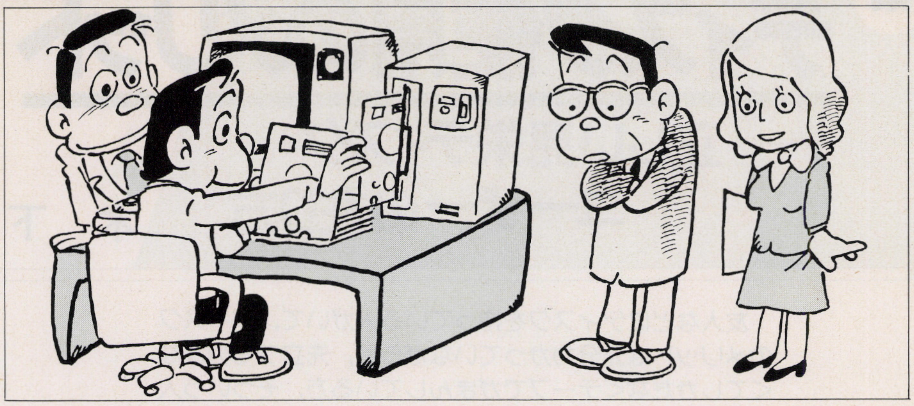
いでしょう。

しかし，人間というのは欲の深い動物で，ディスクに慣れてくると，今度はそれさえも遅く感じられるようになるんです。そこで見栄ディスクユーザーは，たとえ数秒のロードやセーブの間でも，タバコを取り出して１本吸うなり，コーヒーを飲むなりして，イライラした顔で，

「遅いなー」

とつぶやきましょう。

スピードと次に当然あげられることは，

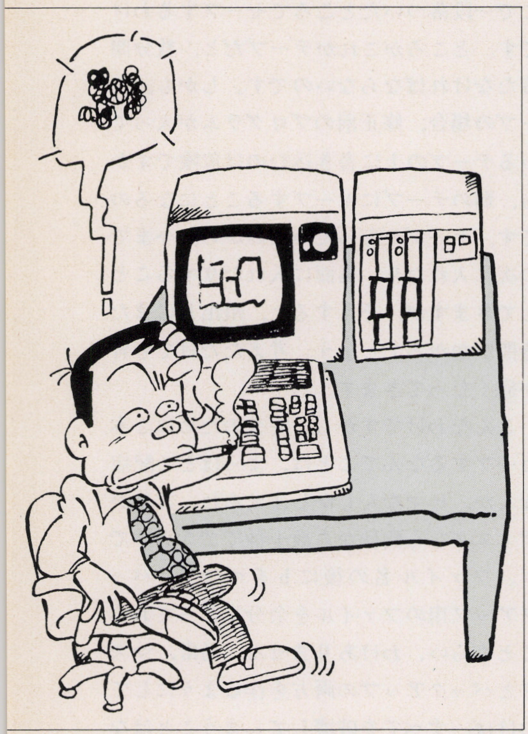


記憶容量が大きい

ということです。

ＦＭシリーズの場合，５インチフロッピで320Kバイト／ドライブ，８インチフロッピで１Mバイト／ドライブがあります。実用性を考えると，テープの場合，１本には表裏の２本のプログラムしか入りませんね。

しかし，ディスクの場合は，プログラムやデータごとにつけられるファイル名によって，１枚のディスケット上に複数のプログラムがあっても，指定したプログラムを選択してロードしたりできるのです。ですから320K，1Mという容量の許す限り（実際はファイル数などにも制限があります），同一のディスケット上にいくつものプログラムやデータがおけるのです。



記憶容量が大きいということは，プログラムがいくつもセーブしておけるという以外に，大規模なファイルを作ることができるということにもつながります。ファイルについてはあとで述べましょう。

記憶容量もスピードと同時に，慣れてくると少なく感じられるものです。そこで見栄をはりたい方は，８インチの両面倍密度ディスケットを２箱買い込んできましょう。そして付属のラベルに

『WM　etc.　—1』

『Languages』

『SEC，U，I　etc.』

『SCREEN　L.』

などという，意味がわかるようなわからないようなことを書いておくのです。

ここで注意することは，

①決してきれいに書かないこと

②ラベルにビッシリ書くなんてことはしないこと

です。さんざん使っていくと，フェルトペンで書き直すためラベルは汚くなっていきますし，入っているファイル名を全部書くなどということはせず，代表的または総称的なことを書くようになります（本当は，ラベルはきれいに書いたほうがいいのですよ）。

そして，この２箱のディスケットを常に持ち歩き，幸運にも８インチのディスクドライブを見つけたら，あれでもないこれでもないと自分の20枚のディスケットをドライブにとっかえひっかえ突込み，最後に，

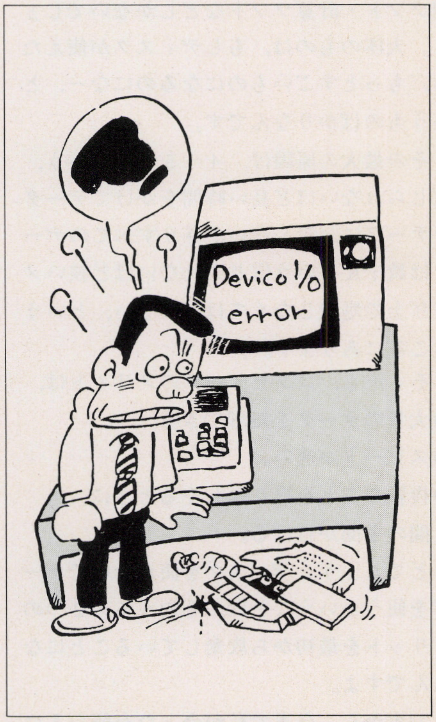
「あー，そういやーあれはハードディスクに入れっぱなしだった。」

といって，その場を立ち去りましょう。

さて，次なるメリットは，テープに比べて，

エラーが起こりにくい

ということです。テープを使ったことのある人はよくご存知でしょうが，テープというのはデータレコーダやテープレコーダによって，相性があるということです。Ａ君が自分のレコーダでセーブしたプログラムを，Ｂ君がロードしようとしてエラーが出て苦労する，というようなことがよくあるでしょう。音量や音質が問題なんでしょうが，とにかくセーブ時とロード時のレコーダは，同一のほうがよいということはいえるようです。しかし，数分間も待ったあげくエラー



ーが出てしまったり，さらに結局はどうしてもロードできなかったりとなると，腹が立ちます。胃かいようにならないためにもディスクを使いましょう。

高級ソフトが使える

というのも，大きなメリットでしょう。市販されているゲーム以外の実用ソフトでは，テープのものはほとんど見かけませんね。だいたい，ン，コレはイイ！と思うものはディスクベースです。

ま，それも当然といえば当然ですね。ゲームを除けば，会計ソフト，ワープロなどは多くのデータを扱うことが目的となり，ディスクでなければ事実上不可能なのです。ある程度のデータは，後述するシーケンシャルファイルとしてテープに記憶させることもできますが，大量には無理ですし，時間がかかって実用にはなりません。

それにゲームにしても，シミュレーションゲームやアドベンチャーゲームなどのように，大量の数値，画面のデータが必要なものや，ピコピコゲームでも多くの場面が設定されているものなどには，ディスクは不可欠ですね。

それから，画面で絵を描くお絵描きソフトにしても，画面データをテープに入れたりしては日が暮れますし，画面の合成，一部切り出しなどを考えても，実用上大きなメリットをもたらす機能は，ディスクならではです。みなさん，もうお気付きかもしれませんが，前号のOh！FMの表紙，あのサンタクロースがサーフィンを持ってるヤツは，このボクが描いたんです。あの表紙がかなりウケまして，多くの励ましのお言葉をいただきましたが，あれもディスクがなければ事実上は書けないものです（アンケートはがき，どうもありがとうございました。おかげでボクも見栄をはることができました。㊍)。

話がだいぶそれてしまいましたが，とにかく，そういうことなんです。みなさんの中にも，

「へー，君，パソコン持ってるの。すごいねー。」

と言われていい気分でいると，

「ところでパソコンで何ができるの？」

と聞かれて，まさかピコピコゲームを見せるわけにもいかず，返答に困った人がいるでしょう。

こんなとき，ディスクがあって英文ワープロでもお絵描きツールでも見せたら，

「なーんだ，結局ゲームマシンじゃんか。」などと言われなくて済みます。

そして，なんといっても大きいのは，ディスクを使うと，

OSが使える

ということです。

FMシリーズで使えるOSでメジャーなものとしてはFLEX，OS-9などがあげられます。OSとは何か？ということを書き始めるとキリがないので，ここではOSが使えると，どういうメリットがあるのか？ということを述べていきましょう。

まずは，様々な言語のコンパイラ，インタプリタが使えるということです。みなさんも，コンピュータ言語にはBASIC以外にもPASCAL，C，FORTRAN，COBOLなどの言語があるということはご存知でしょう。これらの言語は事実上，OSがなければ使用できません。したがって，ディスクがなければ使えないわけです。BASICに飽きた方は，これらの言語にトライしてみるのもよいでしょう。

また，OSが使えると，機種間の隔りがなくなります。つまり，同じOSが走っているマシンは，OS上のソフトならほとんどそのまま共通に使えるということです。全く共通にとは言えない面もありますが，そもそもOSというのはそのためにある，と言っても過言ではないでしょう。ディスクユーザーでも，OSとその上で走るツールなどが使いこなせるようになれば，一人前です。

さて，FM見栄ディスクユーザーのあなたは，話の中でとにかく

「OS-9はすごいよ。」

を連発するように心がけましょう。またCコンパイラに関しても

「ホワイトスミスのオブジェクト生成効率もすばらしいけど，BDSの手軽さも捨てがたいなー。」

と言えたら，エライものです。そして，さらにディスクのメリットを述べるならば，今までも少しずつ言ってきたように

データベースが作れる

ということです。データベースについても説明しようとするとキリがないので，簡単に説明しましょう。とにかく，大量のデータを記憶しておき，必要なときに必要なデータをとり出して処理したり，記憶させたりするものです。

データベースの基本となるのがランダムアクセス，つまり，ディスク上の任意の位置にデータの読み書きができることです。ディスク上には，本体のメモリのアドレスと同じように位置を表すための数値があって，いつでも好きな所をアクセスできるようになっているのです。

このために，ディスクをあたかも，スピードの遅いメモリのようにして使うことができるんです。BASICにも，ランダムファイルというのが用意されていて，データを256バイト単位で，ディスクの上の指定した位置（レコード）に書き込めるようになっています。このランダムファイルによって，会計ソフトやシミュレーションゲーム，住所録，恋人リサーチなどが可能になるわけです。住所録などは1人分の住所を，恋人リサーチなどは1人分の情報を，1レコードに書き込んでおけばいいわけです。こうしておけば，電源を切っても，レコードナンバというランダムファイルのアドレスのようなものを指定すれば，いつでもそのデータを取り出すことができるわけです。

さて，テープでも，データを外部に記憶させることができることはご存知でしょう。テープでできるのは，シーケンシャルファイルというファイルのみです。テープという物理的特性を考えても，ディスクのランダムファイルのようなことはできないのはおわかりでしょう。テープの場合は，書き込む位置までは指定できないでしょ？　つまり，テープだと，行ってしまったらサヨーナラなわけです。書き込む位置はテープの流れがすべてを決めてしまいますよね。ですから，書き込み始めたら最後まで連続して書き込んでやらなくてはいけないし，読み出すときは，最初から読まないといけないのです。このため，テープによるシーケンシャルファイルは，よほど時間的に余裕のある場合でなければ実用的ではありませんね。

ところで，ディスクにもシーケンシャルファイルはあります。ディスクのシーケンシャルファイルは，ディスクの中にテープ

のような帯状の記憶領域があると考えてください。ただし，スピードは全然違いますよ。テープで数分かかるものはディスクでは数秒です。これなら実用的ですね。ワープロのデータなどはシーケンシャルファイルを使うのが普通でしょう。

それに，ディスクにおけるシーケンシャルファイルには，もっと便利な利用法があります。みなさん，ASCIIセーブというのをご存知ですか？　セーブするときに，

SAVE"ファイル名"，A

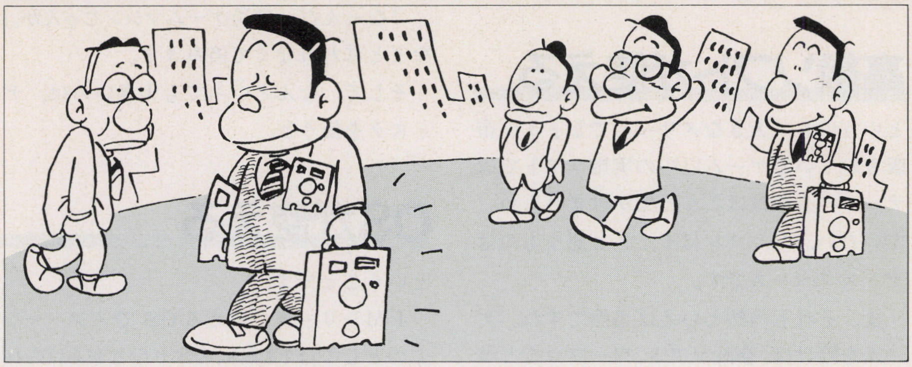
とやるやつですよ。こうしてセーブされたファイルは，プログラムとしてロードすることもできますし，プログラムの1行1行をシーケンシャルファイルの文字列データとして読み出すこともできるんです(これはテープでもできますが，めんどうで時間がかかります)。また逆に，シーケンシャルファイルとして，

"100　PRINT　A"

などという文字列を書き込んでおくと，それをASCIIセーブされたプログラムとしてロードすることもできるんです。

これを使えば，DATA文自動作成プログラムや，変数検索プログラムなど，プログラム自体を処理するプログラムも簡単にできます。

さて，以上のことを考えなくても，とにかく，ディスクを持っていることは，



カッコいい

んです。テープを持っていても，音楽テープと変わりありませんが，ディスクを持っていれば明らかに，コンピュータを使いこなしている人，それも，かなりのレベルの人ということがわかります。それに，あのディスケットのデザインはなかなかのもので，初めて手にしたときは，それをながめているだけでうれしくなるものです。ま，しかし，コンピュータ＝暗いという，あまり感心しない，しかし，しごく当然なイメージが一般に定着している現在は，ディスクを持ち歩くより，音楽テープを持ち歩いていたほうがいいかもしれません。

とにかくディスクはイイものです。帰りに飲みに行こうかなって考えているあなた，そんなお金があったらディスクを買いましょう。でも，ボクは飲みに行きます。もうすでに持ってますから(見栄はっちゃった)。

ディスクおもしろプログラム

踏み切り(FM-7,8,11)

つまらないギャグですが，2ドライブ以上のディスクドライブがつながっている方がこのプログラムを実行すると，踏み切りの味が楽しめます。なお，FM-8で倍速でない方はループを少々，減らしてください。ドライブにディスケットを入れなくても構いませんが，入れて実行したほうがより効果的な音が出ます。

```
40 FOR A=0 TO 1:POKE &HFD1D,A+128:POKE &HFD18,10:BEEP 1:FOR I=1 TO 100:NEXT:BEEP
 0:FOR I=1 TO 500:NEXT:NEXT:GOTO 40
```

フォーマットもどき(FM-7,8,11)

マシン語なしでも，また，ディスク関係の命令なしでもこのようにフォーマットの音は出せます（もちろんフォーマットは行われません)。特に音のうるさいドライブをお持ちの方は，リアルだと思います。これをSTARTUP にして友人を驚かすのも良いでしょう。

```
40 PRINT"Now formatting･･･":POKE &HFD1D,128:POKE &HFD18,10:FOR T=1 TO 39:POKE &H
FD18,&H5A:FOR I=1 TO 1500:NEXT:NEXT:PRINT"FORMAT FINISHED"
```

音だけFM-7(FM-11)

FM-11でこれを実行すると，あのFM-7などで薄型ディスクドライブを使っている気分が味わえます。まあ，これはFM-7がうるさいのはドライブ自体に問題があるのではなく，ソフトで解決できることの証明でもあるわけです。

```
40 POKE &HFD1D,128:FOR I=1 TO 10000:POKE &HFD18,10:FOR T=1 TO 1500:NEXT:POKE &HF
D1B,39:POKE &HFD18,&H1A:FOR T=1 TO 2200:NEXT:NEXT
```

悲鳴

このプログラムを実行してはいけません。万一，実行でもしようものなら，精神衛生上（ディスクドライブにも）良くないサウンドが味わえます。このサウンドは，FM-7用の薄型ドライブのセーブ時等に発生する音とはひと味違います。また，最初のR＝9をR＝8，R＝10等とすると，ディスクドライブ次第で異なる味がします。もう一度繰り返します。決して実行しないでください。

```
40 R=8:POKE &HFD1D,128:POKE &HFD18,R:FOR I=1 TO 10000:POKE &HFD19,0:FOR T=1 TO 1
500:NEXT:POKE &HFD1B,39:POKE &HFD18,R+16:FOR T=1 TO 2200:NEXT:NEXT
```

©1983 T. Tsuruoka

特集 ディスク活用術

FMシリーズにつながるフロッピディスクカタログ

パソコン，ワープロ専用機をはじめてとして，フロッピディスクの進出は目ざましいものがあります。お陰で，価格もかなり低下して，とても手に入らないと諦めていた多くのユーザーにも，身近なものとなってきました。また，最近発売されたパソコンには，5インチ，3インチあるいは3.5インチ等のディスク組み込み型のマシンも多く出現し，ディスクが当たり前の時代になりつつあります。

本誌では，Vol. 2に「各社フロッピディスクドライブユニットの比較検討」を掲載いたしましたが，その後，多くのメーカーが仕様・価格の変更を行ったり，新機種を発売するなどしていますので，今回，新たにFMシリーズにつながるディスクカタログとして集めてみました。

富士通純正のフロッピディスクドライブユニット（以下FDUと略記）は，現在5.25インチのミニFDUと8インチの標準FDUのみですが，3インチ，3.5インチのコンパクトFDUを発売あるいは発売予定のメーカーもあるようです。

また，従来，FMシリーズ用として発売されていた5インチFDUは，FDUの中にフロッピディスクコントローラが基板とともに組み込まれている設計のものがほとんどでしたが，マクロシステムズのFD-7のように，FM-7のI/Oスロットに差し込むフロッピディスクインタフェース内に組み込まれているものも出はじめました。I/Fカードとドライブを別々に購入するときは，ディスクメーカー指定のものを使用したほうが安全といえます。

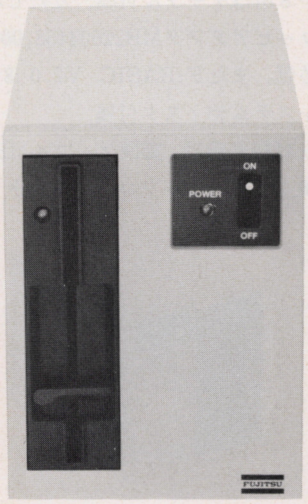
なお，F-BASICシステムディスケットは，富士通のMB27603(V2.0 8インチ付き)以外すべて別売となります。

○F-BASIC　V2.0（5″）7,000円
○F-BASIC　V3.0（5″）7,000円
○F-BASIC　V4.0（5″）7,000円

また，寸法の記載は，幅×高さ×奥行の順です。

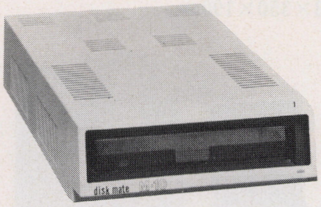
●MB27607（富士通）

FM-7，8用ミニシングルFDU
定価98,000円（MB27608増設時167,000円）ケーブル付属
FM-7接続時には，ミニフロッピI/Fカード（14,800円）が必要
FM-8接続時には，ミニフロッピアダプタ（17,000円）か，システム拡張ユニット（133,000円）が必要
135×180×270mm



●disk mate M10-2（アイテム）

FM-7用ミニシングルFDU
沖電気GM3305ドライブ使用
定価69,800円
ケーブルはオプション（5,000円）
フロッピインタフェース付属
170×70×315mm

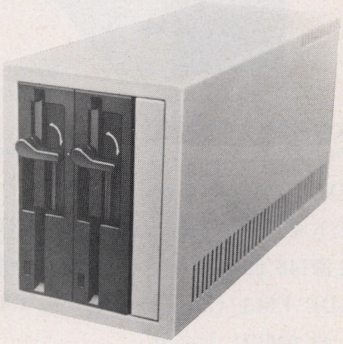


●disk-80FⅡ（アイテム）

FM-7，8用ミニデュアルFDU
松下通信工業JA-551-2Nドライブ使用
定価128,000円，シングル時98,000円
接続には，ケーブル（5,000円）とdisk-I/O-F（15,000円）が必要
120×170×350mm

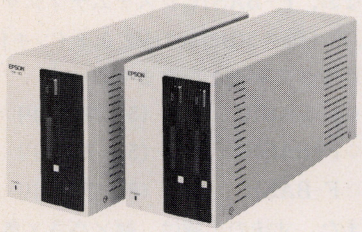
●THIN LINE/FM（亜土電子）

FM-7用ミニデュアルFDU
YE-DATA YD-580ドライブ使用
定価120,000円
ケーブル付属
インタフェースカードはオプション（価格未定）
130×170×300mm



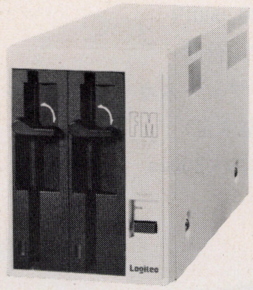
●TF-10(エプソン)

FM-7，8用ミニFDU
エプソンSD-340ドライブ使用
定価シングル85,000円
　デュアル125,800円
FM-7接続時には，ミニFDインタフェースカード＃9040（12,000円，ケーブル付属）が必要
FM-8接続時には，ミニFDインタフェースカード＃9010（15,000円，ケーブル付属）が必要。ただしシステム拡張ユニットMB26001がある場合にはケーブル（5,000円）のみ必要
120×165×350mm



●Logitec K-305FM(関東電子)

FM-7，8用ミニデュアルFDU
TEAC　FD-55Bドライブ使用
定価128,000円
FM-7接続時には，I/FカードK-305-12（13,800円）とケーブル（4,000円）が必要
FM-8接続時には，I/OボードK-305-11（13,000円）とケーブル（4,000円）が必要
122×181×360mm



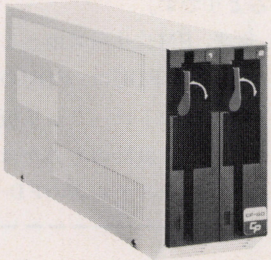
●KD-290D/FM(工人舎)

FM-7，8用ミニデュアルFDU
松下通信工業JA551-2Nドライブ使用
定価148,000円（シングル時98,000円）
ADP-FM1（15,000円）とケーブル（3,000円）が必要

430×76×297mm

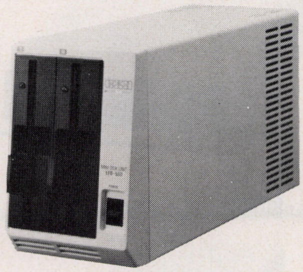
●CF-80FM(湘南電送)

FM-7，8，11用ミニFDU
TEAC　FD-55Bドライブ使用
定価CF-80FMS（シングル）
　98,000円
　CF-80FMD（デュアル）
　128,000円
ミニフロッピアダプタMFP-22603（13,000円，ケーブル付属）が必要
88×172×350mm



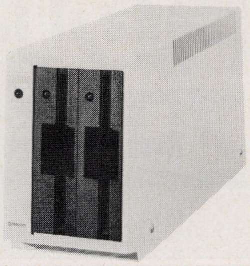
●LFD-550FM(東京電子科学)

FM-7，8用ミニデュアルFDU
TEAC　FD-55Bドライブ使用
定価148,000円
I/Oボード・ケーブルセット（15,000円）が必要
123×350×190mm



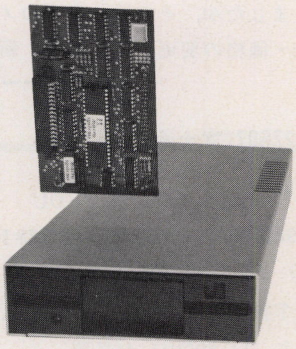
●NH-200F-FM(ニデコ)

FM-7，8用ミニデュアルFDU
アルプス電気FDD-2225ドライブ使用
定価138,000円
インタフェースボックスFDC-F01（14,800円，ケーブル付属）が必要
130×185×352mm



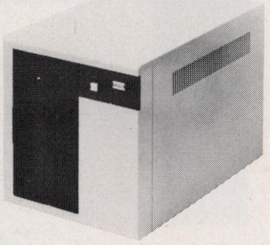
●FD-7(マクロシステムズ)

FM-7用シングルFDセット
東芝ND04Dドライブ使用
定価99,900円
増設用/5″MACRO-FD5（74,800円）
　8″MACRO-FD8（149,000円）
　3″MACRO-FD3（74,800円）
　3.5″MACRO-FD4
　（予価74,800円）
MACRO-7-5（フロッピディスクI/F）付き，ケーブル付属
50×150×300mm



●MB27603(富士通)

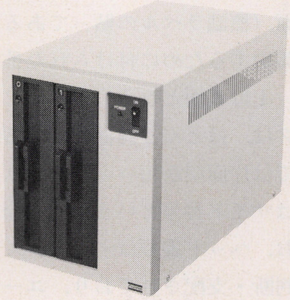
FM-8，11用標準デュアルFDU
定価440,000円
FM-8用はケーブル付属，ただし標準フロッピアダプタMB22604(58,000円)が必要。またF-BASIC V2.0 8インチシステムディスク付属
FM-11接続時は，変換ケーブルMB26532(8,000円)，標準フロッピコントロール



カードMB22415（30,000円）が必要
240×264×370mm

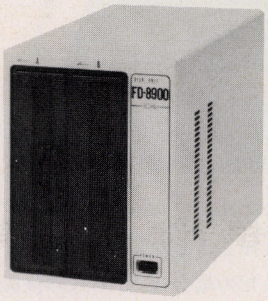
●MB27613（富士通）

FM-11用標準デュアルFDU，およびFM-8の標準FDU増設用
定価350,000円
FM-11接続時には，ディスクケーブルMB26526（8,000円）と標準フロッピコントロールカードMB22415（30,000円）が必要
FM-8の標準FDU MB27603の増設用には，増設ケーブルMB26534（8,000円）が必要
240×264×370mm



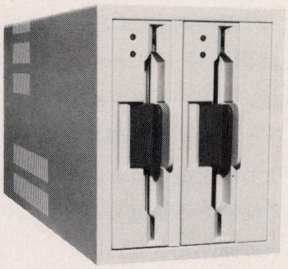
●FD-8900（アイシーエム）

FM-11用標準デュアルFDU
YE-DATA　YD-180-1600ドライブ使用
定価298,000円，
シングル時198,000円（FD-8901）
ケーブル付属
富士通純正標準フロッピコントロールカード（30,000円）が必要
202×267×388mm



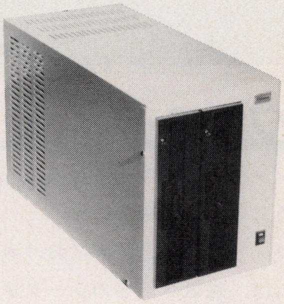
●THIN LINE 8/FM-11（亜土電子）

FM-11用標準デュアルFDU
YE-DATA　YD-180-1600ドライブ使用
定価268,000円
ケーブル付属
富士通純正標準フロッピコントロールカード（30,000円）が必要
155×240×450mm



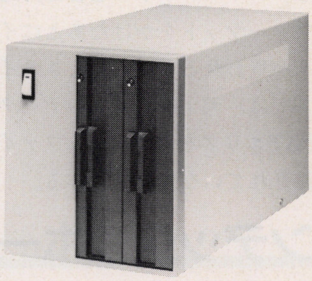
●FDU-8FM1（エヂソン・ハウス）

FM-11用標準デュアルFDU
YE-DATA　YD-180-1600ドライブ使用
定価262,000円（シングル182,000円，組立キット3,000円引き）
ケーブル付属
富士通純正標準フロッピコントロールカード（30,000円）が必要
132×190×370mm



●FD-11（ラトックシステム）

FM-11用標準デュアルFDU
YE-DATA　YD-180-1600ドライブ使用
定価298,000円（シングルFD-11S：198,000円）
ケーブル付属
富士通純正標準フロッピコントロールカード（30,000円）が必要
200×221×380mm



●問い合わせ先●

富士通㈱	全国スカイラブ	
㈱アイシーエム	〒556	大阪市浪速区戎本町1—7—1 ☎06(647)2008
㈱アイテム	〒251	藤沢市南藤沢8—1—202 ☎0466(27)1668
㈱亜土電子工業	〒101	東京都千代田区外神田2—4—6 ☎03(253)8303
エヂソン・ハウス	〒141	東京都品川区東五反田2—3—10 ☎03(440)4010
エプソン販売㈱	〒160	東京都新宿区西新宿2—4—1 新宿NSビル　私書箱6109号
関東電子㈱	〒101	東京都千代田区外神田2—15—2 ☎03(257)6245
㈱工人舎	〒231	横浜市中区松影町2—7—21 ☎045(662)0688
湘南電送機器㈱	〒252	藤沢市高倉12　湘南台中央ビル ☎0466(45)3535
東京電子科学機材㈱	〒101	東京都千代田区外神田2—4—4 ☎03(251)3271
㈱ニデコ	〒101	東京都千代田区神田松永町1 ☎03(253)0761
マクロシステムズ㈱	〒101	東京都千代田区神田1—7—6 ☎03(251)5148
ラトックシステム㈱	〒550	大阪市西区江之子島1—6—2 ☎06(445)1657

特集 ディスク活用術

ディスク用語集

アクセスタイム

データの書き込み，または読み出しを行わせるに要する時間をいう。

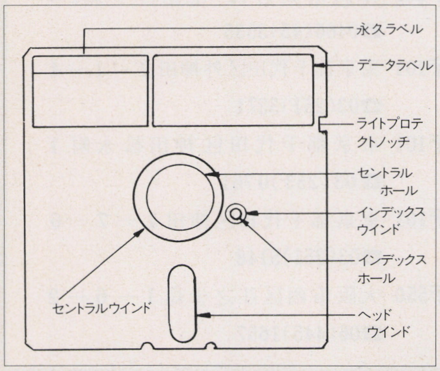
フロッピディスク装置におけるアクセスタイムは，磁気ヘッドのシークタイムと，回転待ち時間(サーチタイム)，さらに情報の書き込みまたは読み出し時間を加えたものになる。

イニシャライズ

ハード，ソフトにかぎらず，使用開始時点でシステムをある安定状態にセットアップすることをイニシャライズ(初期設定)という。たとえばBASICプログラムをRUNさせると，変数の値は0にイニシャライズされる。ディスケットの場合は，通常アドレスフォーマットをかけたあとにイニシャライズが行われるが，このときのイニシャライズでは，ディスケットのデータエリアや，FATエリアをクリアし，ファイルが全く登録されていない状態を作る。

インデックスホール

フロッピディスクの各トラックの開始位置を検出する穴のこと。ディスクジャケットには1個所，直径5㎜程度の穴があいている。一方，マイラベースのほうにも1個所穴があいている（インデックスホール）ため，1回転するたびに1度だけ光がこの部分を通過することになる。これを検知することにより，フロッピの物理的なスタート位置を知ることができる。なお，8インチフロッピの場合は，この穴の位置により，片面か両面かの判定も行っている。



FCB (File Control Block)

ファイルコントロールブロックは，DOSがディスクとの間で入出力動作をするために必要なすべての情報をもったブロックのことである。1つのファイルごとに1つのFCBが作成される。

FCBはフロッピのディレクトリエリアに置かれ，必要に応じて主記憶上に引き込まれる。FCBの内容はOSにより異なるが，通常，装置番号，ファイル名，ファイル名拡張子，レコードサイズ，シーケンシャルリード／ライト時の各時点でのレコード番号などが記録される。その他システムが内部で使用する領域もあり，この部分はユーザーは操作できない。

片　単

フロッピディスクのタイプを省略して呼ぶときの言いかた。

片単……片面・単密度
両倍……両面・倍密度
両倍倍トラ……両面・倍密度・倍トラック

などと呼ばれる。なお，これらフロッピのタイプによって，価格は当然違ってくるが，これは，メーカーが品質を保証しているかどうかだけであり，すべて同じように作られている。筆者は両面倍密度タイプのフロッピを，両面倍密度倍トラックタイプのドライバに入れて使用しているが，おかしくなったことは一度もない。Appleの片単フロッピをひっくり返して両面使う，というのは有名なテクニックである。Oh! Secoism!!

サーチタイム

ディスク装置で，磁気ヘッドが指定されたトラックの上まできた後，指定されたセクタが来るまで回転待ちする時間。平均サーチタイムは，ディスクが1回転する時間の半分である。

サイド番号

ヘッド番号ともいう。フロッピディスクの表面（標識ラベルの貼ってある面）と裏面を識別するために付けられた番号。0は表面，1は裏面となる。現在のフロッピディスクは，ほとんど両面型であるが，8インチでは片面型が用いられており，この場合にはサイド番号というものはない。

なお，ヘッド番号はすべてシステム機能にて判断・支配されており，ユーザーが利用にあたってこれを意識する必要はない。

シークタイム

フロッピディスク装置とのデータの入出力を行う場合，磁気ヘッドが指定されたトラックまで移動するのに必要とする時間のことをいう。一番外側にあるヘッドが一番内側のトラック位置まで移動する時間が，最大シークタイムである。

磁気ヘッドがこのように大きな移動ばかり行うと，主記憶と外部記憶装置との転送効率が悪くなるので，通常はシステム側でなんらかの最適化を行っている。

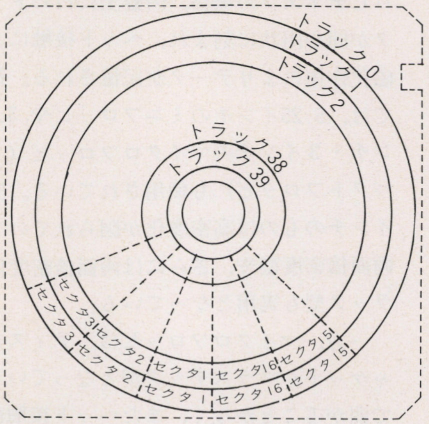
セクタ

フロッピディスクとのデータの入出力を効率よく行うために，トラック上の記録部分を区分することが行われる。この区分された書き込みや読み出しの最小単位をセクタと呼ぶ。たとえば16セクタタイプのフロッピの場合は，トラックの円が16分割されていることを意味する。

セクタ分けは，フロッピの型により，また各メーカーの設計思想により異なるが，一般には，8，16，26セクタのどれかにな

っている。1セクタあたり何バイトデータが書き込めるかも決まっていて，128バイトとか，256バイト，512バイトなどの値となっている。ちなみにF-BASICでは，16セクタ／トラック，256バイト／セクタとなっている。

F-BASICのディスク構造



セクタの内容は，ユーザー書き込みデータだけで構成されているのではない。正しくその位置を知るための情報も付加されている。たとえば，タイミング決定のための同期パターン，セクタの番号を示すIDフィールドなどである。

1枚のフロッピにユーザーが書き込めるデータ量は次のとおりとなる。

データ量＝サイド数×トラック数×セクタ数×1セクタあたりのバイト長

ディレクトリ

フロッピ上に作られたファイル名などを管理するエリア。ディレクトリエリアには実はFCBがそのまま出力される。

ただし，通常ディレクトリといわれているものや，CP/MやMS-DOSのDIRコマンドが対象としているのは，単にディスクに収容されたファイル名の集合を意味することが多い。

ディレクトリを見るというのは，どういうファイルがディスクに収容されているかを調べることである。

DOS(Disk Operating System)

単にDOSというと，大型コンピュータやミニコン用のディスク装置を外部記憶と想定したオペレーティングシステムのことをいう。パソコンレベルでは，通常，フロッピディスクを外部記憶としたOSのことであり，とくに区別するときにはFDOS（Floppy DOS）と呼ばれたりする。

OSもまた，各種プログラムの集まりであるが，業務プログラムと違って，オフィスの特定の仕事を行うものではなく，プログラム開発時間の短縮や，コンピュータの有効利用をするための手助けといった目的で使用される。

本来，OSは制御プログラムと処理プログラムよりなり，これらの中には，マルチジョブ機能やFORTRANなど言語処理プログラムなどは当然含まれる。これに対し，マイコンのシステムにおいては，制御プログラム部をOSと称していることが多い。また，ときにはその中のさらに基本機能であるモニタ（あるいはスーパバイザ）部のみをOSという場合もある。

しかし，一般に名の知れたOSでは，モニタプログラム部と，フロッピディスクを中心とした簡単なファイル管理機能を含めてFDOSと呼んでいる。DOSの目的は，無論，プログラム開発や運用上の効率アップであるが，パソコン用DOSの場合は，むしろそれよりも，共通DOSを媒体にしていろいろな言語やアプリケーションプログラムが走る，というソフトウェア上の共通バスといった目的意識のほうが強い。

有名なFDOSとしては，ディジタルリサーチ社のCP/M，マイクロソフト社のMS-DOS，またOS-9やFLEXなどがある。

ドライブスピンドルホール

フロッピディスクの中心にあいている穴のこと。セントラルホールともいう。この穴にＦＤＤの回転軸が入り，フロッピディスクが固定され，回転する。

トラック

フロッピディスクの記録場所を表す名称。読み書きするためのヘッドをある位置に固定させて，フロッピディスクを1回転させると，1本の円ができるが，これをトラックと呼ぶ。トラックとトラックの間には，データ誤りを起こさせないために適切な間隔が必要で，そのためトラック数は、フロッピディスクの直径に依存するある値に決定される。通常5インチ型では片面で0～39の40トラック，8インチ型では0～76の77トラックである。いずれの場合も外側から0，1，2……という番号になっており，両面型の場合は，当然トラック数は倍になる。

なお，最近5インチ型フロッピに倍トラック型がでてきていて，この場合は片面で80トラック，両面で160トラックとなる。

ノッチ

フロッピディスクの書き込み禁止を行うもの。5インチフロッピの場合，ディスクジャケットの一端がコの字型に切り落とされている。通常の使用では，このままにしておくが，完成したプログラムをフロッピに格納し，誤操作などにより，これを消したりすることを防ぐ場合には，ここにシールを貼り付ける。これにより，フロッピディスク装置側で機械的に書き込み禁止状態を設定する。なお，8インチフロッピにも書き込み禁止ノッチのあるものがあるが，5インチフロッピと使用方法が全く逆で，シールをはがすと書き込み禁止となる。

ファイル

特定の目的にしたがって計画的に集められ，組織化された情報の集まりをいう。

これらは大別すると，データファイルとプログラムファイルになる。データファイルは編成法により，さらにシーケンシャルファイルとランダムファイルに分けられる。

カセットテープ上のファイル形式はシーケンシャルファイルのみである。シーケンシャルファイルは，大量のデータを格納できるが，特定のデータだけを取り出すのは苦手である。ディスク上では，シーケンシャルファイルのほか，ランダムファイルを作ることができる。ランダムファイルは，どのレコードも同様な速度で読み出すことができるが，記憶効率は少し落ちる。また，順次読み出しを行うときの速度効率は保証されない。

プログラムファイルは，フロッピディスク上ではシーケンシャルファイルとして格納される。したがって，これをシーケンシャルデータとして読み出すことができる。

またBASICにてセーブしたプログラムファイルは，PRINTやINPUTなどの予約語がすべて中間コードという特殊な1バイトあるいは2バイトの形で格納される。これを禁止して，CRT上で見る文字のコード(ASCIIコード)そのままでセーブすることもできる。このときできるファイルは，そ

の記録形式上，ASCIIファイルと呼ばれる。

FAT(File Allocation Table)

フロッピにファイルをセーブした場合，そのファイルが実際にフロッピ上のどこに書かれているかの情報を記録するテーブルである。FATはまた，フロッピの未使用領域も示す。フロッピディスクへのデータ読み書きの最小単位はセクタであるが，このセクタ単位にテーブルを作ると，テーブルサイズが大きくなり，また効率的ではないので，実際には複数セクタ（FMシリーズでは8セクタ）単位に――これをクラスタと呼ぶ――テーブルを作り，

①そのクラスタが未使用かどうか

②使用中の場合，その領域は最終クラスタか

③最終クラスタでない場合，次はどのクラスタに続くのか

などの状態を記録する。

FATは，F-BASICではDSKINIの実行で初期化される。一般に，初期化の際には，FAT自身が収容されているクラスタや，キズがついていて使用できないトラックがあると，そのクラスタに相当するFATに使用中というフラグをたてる。しかし，F-BASICではこれらは行われない。

フォーマッティング

フロッピディスクは，製造されたままでは，何も書かれていない真っ白な紙のようなものである。だからこれを実用にするためには，指定したトラックの指定したセクタに正しく位置決めできるように，またフロッピに対するなんらかの管理を行うために，目次のようなものの領域を割り当てておかなければならない。これを行うことをフォーマッティングという。

フォーマッティングを行うと，当然ユーザーの使用可能領域は少なくなる。だからフロッピディスク容量の説明として，しばしばアンフォーマット時○○○Kバイト，フォーマット時○○○Kバイトという表現が行われる。

フロッピディスク

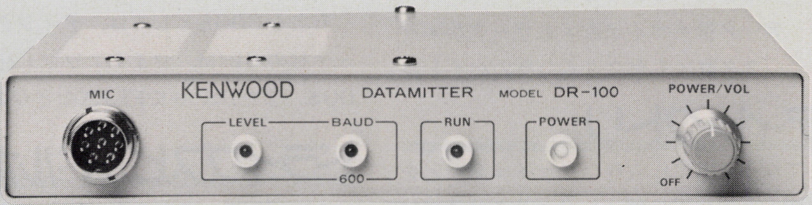
単にフロッピとか，フレキシブルディスク，あるいはディスケットともいわれている。ディスケットは正確にいうとIBMにおける商品名である。

薄い円盤状のマイラベースの両面に磁気コーディングが施してある。これを黒いディスクジャケットに収めている。標準フロッピディスクの場合，円盤上に77のトラックが同心円状に刻まれ，ヘッド接触による磁化作用によりデータが記憶される。最近では，5.25インチのミニフロッピや，3.5インチ・3インチのマイクロフロッピ（コンパクトフロッピ）も使用されている。5.25インチのものは高密度化が図られていて，両面倍密度型や，さらには両面倍密度倍トラック型も実用となっている。

一方，マイクロフロッピは，ディスクジャケットがプラスチック型となっているので曲がることがなく，またヘッド接触部にはシャッタが付いているなど，信頼性の向上が図られている。

現在，FMシリーズで最も一般的に使用されているのは，5.25インチのミニフロッピで両面倍密度型のものである。

アマ無線機とFM-7を結ぶ DR-100



トリオ㈱(〒150東京都渋谷区渋谷2-17-5☎03-486-5520)は，このほど，アマチュア無線機とパーソナルコンピュータをつなぐ画像(データ)通信用インタフェースDR-100を，1月下旬より発売すると発表した。

アマチュア無線による交信方法は，従来，音声による交信やモールス符号によるものが主流であったが，パソコンを利用して，電話級，電信級の資格者にも手軽に画像(データ)通信が楽しめるよにと企画された商品である。

対応パソコンは，FM-7，PC-8001，同mkII，MZ-2000の4機種で，それぞれ別売の中継接続器CS-100が必要となる。対応トランシーバは，430MHz帯FMモードのもの。

主な特徴は次のとおり。

- 音声による交信に加えて，手軽に画像(データ)通信が行える
- トランシーバ，パソコンを改造することなく行える
- DR-100のフロントパネルにある8ピンメタルジャックにマイクを接続することにより，従来どおりの交信も可能
- 600ボー，300ボーの2段切換機能
- マイクラインとデータラインを自動的に切り換える

また，ソフトは，各中継接続器に付属の「レター」(文字や記号などパソコンのディスプレイに作成したデータの送受信プログラム)のほか，FM-7用として，キャンパス(LINE，PAINTなどで作画した絵の送受信)，QSLカード（QSLカードの送受信およびプリントアウト)，コミュニケーション(同一パソコン間のプログラム交換)，将棋(2ウエイゲーム)が用意されている。レター以外はすべて別売で，各4,000～5,500円。

なお，本体DR-100の価格は28,000円，中継接続器CS-100は9,800円。

特集 ディスク活用術

ディスクユーティリティ集

for FM-7 ディスクエディタ「ZAP FM」

平野 剛

ZAP（ザップ）を知っとうや。うちの部落の秘伝だが，このたび大公開。起源は古くAPPLE時代にさかのぼるこの由緒正しきツール，しかしその生いたちについては，禁句なので，口が裂けても言えません，あしからず。

基本的なコンセプトは，ひとくちに言って，ディスクのセクタ修正（バイトごとの書き換え）につきるが，その使い勝手は一味も二味も違う。

もともと，FLEX上で走っていたこのプログラム，今度はF-BASIC上で走らそうというわけで，F-BASIC用の下位ルーチン入出力関係をこれに接続して，ZAP FMというツールにしてみた。これをSimple LogicのFLEX→F-BASIC ファイルコンバータ（これはすごい！）でF-BASICへもってきた，というわけ。

BASICでもFLEXでも，CP/MでもLogoでも，IBMフォーマットで256バイトセクタのディスクなら，なんでも読めて，変更できて，書き込めるというすごいツールなのです。

使い方

使い方は簡単，画面に出てくるとおりにすればよい。

R　リードセクタ

R␣トラックナンバ␣セクタナンバ␣[R]⏎

というように，ブランクを区切りにトラックナンバとセクタナンバ，そしてディスクの裏面を見たいときには“R”を付けてリターンキーを押す。

トラックナンバ，セクタナンバともに16進数で，しかも大文字で入力します。

このRコマンドの機能は，指定したセクタのイメージをバッファに読み込むこと。

Z　ザップセクタ

Z␣[オフセット]⏎

で読み込んだセクタのイメージを持つメモリ上のバッファを1バイトごと書き換える。

オフセットというのは，どこから書き換えるか，そのはじまりを指定する値。セクタの先頭を0として，255番目までのどのバイトから書き換え始めるかを，0～FFの16進数で教える。省略すればオフセット＝0で，先頭からとなる。

Zコマンドを入力すると，指定した書き換え先頭バイトの値と16進とASCII文字が表示されて，何に書き換えるか聞いてくる。たとえば，

Z␣12⏎

と入力すると，

0012-41(A)⇒

となる。これはセクタの13番目（1番目は0000だから）のバイトが現在16進表示で41，ASCII表示で“A”になっていて，さてこれを何に直しますか？と言っている。ここで，入力は4とおり。

まず，ただリターンキーを押すと，次のバイトへ進む。

0～FFまでの16進数を入力すると，そのとおりに書き換える。

シングルクォート（'）を付けてASCII文字で入力すると，その文字に対応したASCIIコードに書き換わる。

0～FFの16進数やシングルクォート以外の文字を入力したときには，書き換え終了とみて，コマンド入力待ちに戻る。

W　ライトセクタ

W⏎

で，バッファ中のセクタイメージをディスクへとコピーする。この命令でセクタナンバの指定ができないことからわかるように，書き込むセクタは読み込んだセクタと同一のもの。すなわち，ZAPは，読み込んで，バッファ上で書き換え，同じところへ戻すことを目的としていて，読み込んだセクタを別へ書き写す，というセクタのコピー機能は持たせていない。これは，もっぱら安全性への配慮がその理由で，ただでさえ危険なこのツールを，ただのファイル破壊装置にしないための充分納得のいく制限であるといえよう。——な～んて言うと，この文体，完全に「間違いだらけのツール選び」みたいでしょ？

Q　クイット

F－BASICへ戻るには

Q⏎

とすればOK。BASICのホットスタートに戻る。

H　ヘルプ

どんなふうに使えばよいか忘れてしまったときに使えばよい。すなわち

H⏎

とすると，各コマンドとその使い方が華麗に現れ，コマンドの使い方を忘れてしまった情けないユーザー（あなたです，そう）を力づけてくれる。まかせなさい。

P　プリンタ出力

リードしたセクタのダンプを画面のみならず，プリンタにも出力したいときに使う。

P⏎

で，プリンタ出力ON，もう一度

P⏎

で，プリンタ出力OFFとなる。

ただし，プリンタに出力されるのは，あくまでリード時のもの。ライト後が見たいときには，もう一度リードすればよい。

と，以上の6つがコマンドのすべて。

LOADM“ZAPFM”，，R

で動かすと，まずコマンドの使い方が親切に出てきて，

Which Disk Drive：

と，「どのディスクドライブの中のディスケットをZAPするの」とたずねてくるから，0か1で誠意ある返事をする。

それからは，上述のコマンドを使ってZAPし，終わったらクイット脱けてBASICへ，というのが使い方のすべて。

注　意

Zコマンドを使ったあとにW（ライト）するのを忘れずに。Zはメモリ上のバッファを書き換えるだけ，Wによって，それをディスクへ書き移さないと修正は完了しない。というよりは修正しかけたとも言えない（図1参照）。

最後に

使い方は簡単だが，これはすごーく恐しい，本当に恐しいツールなんだから。

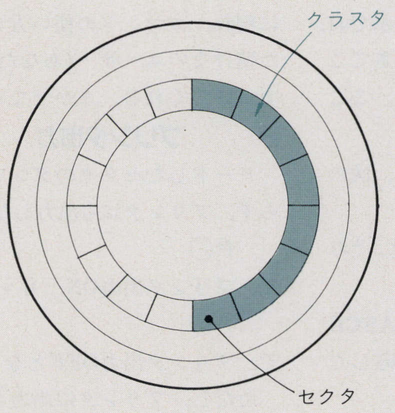
テキストファイルの1バイトくらいならまだしも，F-BASICのディレクトリやFAT，あるいはFLEXのファイルのチェーンを支えている各セクタの最初の2バイト（リンケージバイト）などは，よほどの慎重さと準備と保険がない限りは，手を出さないこと。タフなやつほど無謀なことはしないものだぜ。もし，F-BASICのFATを壊してしまうと，ファイルを読むときにBASICシステムは，そのファイルが書かれているクラスタを見つけだす術をなくしてしまう。そうなると，もうシステムはガチョーンだし（運が良くてエラーメッセージ，普通なら背中のスイッチをやさしく押すしかない），ディスクはメチャクチャ。

でも背中のリセットスイッチをやさしく押してあげるときに，「ハラホロヒレ」とつぶやくことのできた余裕派のあなたなら，きっと修復できるはず。F-BASICのシステムでは，FATを壊しても，ディレクトリと付き合わせればすぐに元の姿を割り出せる（ディレクトリに書いてある開始クラスタナンバを手がかりにする）し，逆にディレクトリを壊しても，FATを見れば，まずファイルだけは再現できないことはない。FLEXだって，チェーンがこみ入っていないうちなら，ディレクトリとリンケージバイトは互いに補いあってくれる。でもこれは，ややこしいパズルになる場合が多い(こんなふうにパズル解きをするときのために，セクタのダンプをプリンタに出力したいと

FLEXのリンケージバイト

F-BASICでは，ディスクのトラックに16個ずつ並んでいるセクタを8つまとめて，クラスタと名付け，これを管理対象の最小ユニットにしている。だから，ディスクにしまえる容量をクラスタより細かくは考えない。ということは，たった100バイトのプログラムでも1クラスタ＝256バイト×8セクタの広さを使ってしまう。ディスクをむだだらけに使うことが時々でてくる。

では，どうしてそんな大きなものを最小単位にするのかというと，



それは管理がたやすくなるから。クラスタ単位なら，両面倍密のディスケットが，クラスタ数でたった156個になってしまう。これを管理しようと思えば，そのクラスタが使われているかどうか，どのくらいまで使われているか(本当は必要ない情報)，そのクラスタがファイルの最後のクラスタか，そのクラスタが使われきっているならその続きはどのクラスタにつながっているのか，などの情報をもったテーブルを持てばよい。それを参照したり，書き換えたりすればよいわけだ。このテーブルが，実はFATと呼ばれているもの。このFATは1セクタに納まってしまう。たった1セクタで記憶容量の管理ができてしまい，速くて便利だ。

一方，FLEXでは，ディスクの記憶容量をたいせつにする。だから管理の対象の最小単位を1セクタにしている。そうすると，F-BASICのようにFATを使っていてはFAT自体が大きくなる(8倍のさらに2倍くらい)し，いちいちFATを参照しながらだと，ファイルへのアクセスがとても遅くなる。なぜなら，フロッピディスクのアクセスタイムの決め手は，トラックのシークタイムだから。フロッピのヘッドがFATのあるトラックとファイルのあるトラックを行ったり来たりしていると，どんどん遅くなる(それなら，FATをメモリ上へ，と考える人はPCユーザーか昔のMS-DOSユーザーの所へ行って，何か不都合が起きるか聞いてください)。

では，なるべくヘッドを動かさないで，次の続きのセクタがわからないか，というわけでFLEXで使っているのがチェーン(鎖)。セクタの最初のバイトと次のバイトの2バイトが，リンケージバイトと呼ばれていて，続きの次のセクタのありか——つまり，次のセクタが存在するトラックナンバとセクタナンバ——が書いてある。これを読めば，なにもFATなど参照しなくても，セクタをたどって速くファイルをアクセスすることができる。ただ，すべてのセクタが少なくとも2バイト（本当は，まだほかにも2バイトの計4バイト）は使えなくなっている（こんなのはF-BASICでむだになっているバイト数に比べれば微々たるもの）し，チェーンというのは，1個所切れると，そこから先が失われるという危険性を伴っている。また，ファイルの更新や書き込みがひんぱんに行われた末には，このチェーンが複雑にトラック間にまたがり，チェーンを利用するメリットであるヘッド移動の少なさが失われる，といったいくつかの欠点も併せもっている。

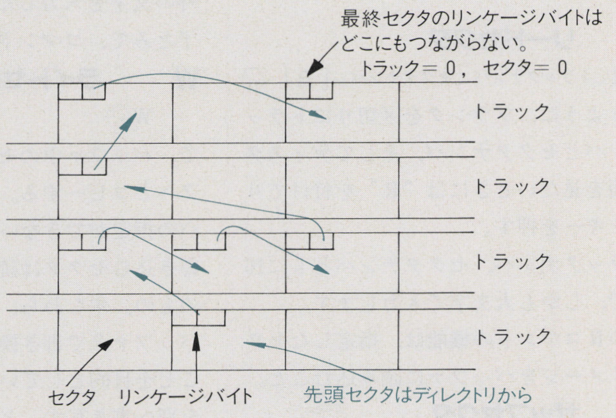
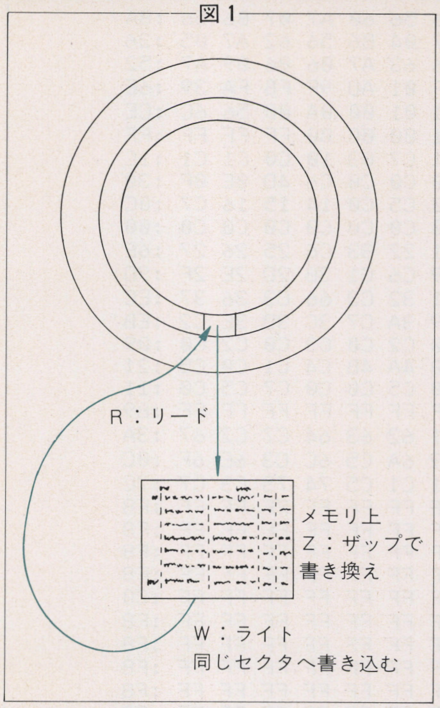


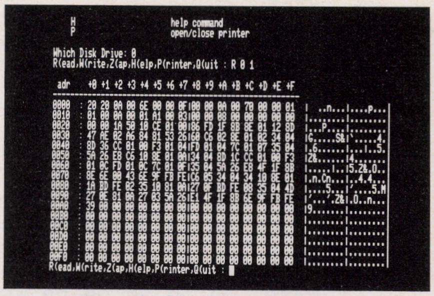
図1



きには，まずPコマンドでプリンタ出力ONにしてから，Rコマンドを使えばよい。画面とプリンタの両方に，トラックナンバとセクタナンバ，そしてダンプが現れる)。

こうしたディスケットの修復にこのZAP FMは使える。だけど，しつこいようだけど，慎重に慎重に（でも，本当を言えば，このZAP FMはこんな使い方のためにあるのです)。自信のない人は，もっぱらセクタの内容を調べるのに使うといい。各システムがどんなディスクファイルシステムを使っているのか勉強するには，直にセクタを見てみるのはとても役に立つから。

また，ブートセクタなど少しのセクタを手早くコピーしてしまいたいときにも，このZAP FMは使える。リードして，ディスケットを取り替えてライトする。これでセクタのコピーができる。もちろん，あらかじめフォーマットすることを忘れずに。



本当に最後に

今号のバージョンは，FM-7用で，8や11では走らない。次号では，必ずやパッチバイトのお知らせをして，8と11のユーザーにもこの劇薬をひろめたいと思っています。乞う御期待！　だから8と11のユーザーも，暇ならダンプリストを打ち込んでおいてね。

リスト1　ZAPFM（入口番地 &H5100）

©1983 G. HIRANO.

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
5100  17 07 30 17 01 98 17 03 :18
5108  85 17 04 F4 17 00 3B 81 :67
5110  52 26 05 17 00 6B 20 F4 :13
5118  81 5A 26 05 17 00 A6 20 :E3
5120  EB 81 48 26 05 17 01 76 :6D
5128  20 E2 81 51 26 03 7E 86 :01
5130  84 81 57 26 05 17 01 51 :F0
5138  20 D2 81 50 26 05 17 07 :0C
5140  0B 20 C9 86 07 17 06 E0 :7E
5148  20 C2 17 06 1F 0D 52 28 :A5
5150  65 61 64 2C 57 28 72 69 :B0
5158  74 65 2C 5A 28 61 70 2C :84
5160  48 28 65 6C 70 2C 50 28 :55
5168  72 69 6E 74 65 72 2C 51 :11
5170  28 75 69 74 20 3A 20 00 :F4
5178  17 06 1D 17 06 26 84 5F :60
5180  39 17 06 20 17 06 49 10 :EC
5188  25 01 0D F7 56 61 17 06 :FE
5190  3F 10 25 01 03 F7 56 62 :27
5198  17 06 09 84 5F 81 52 27 :03
51A0  05 7F 56 63 20 05 86 01 :E9
51A8  B7 56 63 BD 56 13 10 25 :CB
51B0  00 E6 7D 05 AC 27 03 17 :55
51B8  03 11 8E 56 6C 31 89 00 :1E
51C0  FF 17 03 4F 39 17 05 DC :99
51C8  17 06 05 10 25 00 C9 FD :1D
51D0  52 87 4D 10 26 00 C1 FC :19
51D8  52 87 BD 58 1A 17 05 8C :B0
51E0  20 2D 20 00 BE 52 87 A6 :AA
51E8  89 56 6C 17 06 33 17 05 :B7
51F0  7B 20 28 00 BE 52 87 A6 :00
51F8  89 56 6C 81 20 10 2B 00 :27
5200  09 81 7F 10 27 00 03 16 :59
5208  00 02 86 2E 17 06 19 17 :03
5210  05 5A 29 20 3D 3E 20 00 :43
5218  17 05 7D A6 84 81 00 10 :54
5220  27 00 47 81 27 10 26 00 :4C
5228  07 30 01 E6 84 16 00 33 :EB
5230  81 30 10 2B 00 46 81 3A :ED
5238  10 2B 00 1B 81 41 10 2B :53
5240  00 3A 81 47 10 2B 00 0F :4C
5248  81 61 10 2B 00 2E 81 67 :33
5250  10 2B 00 03 16 00 25 17 :90
5258  05 76 10 25 00 1E 4D 10 :2B
5260  26 00 19 BE 52 87 E7 89 :46
5268  56 6C BE 52 87 30 01 BF :49
5270  52 87 1F 10 83 01 00 2A :B6
5278  03 16 FF 5B 8E 56 6C 31 :F4
5280  89 00 FF 17 02 8D 39 00 :67
5288  03 B6 56 61 10 2B 00 08 :B3
5290  17 03 AB 10 25 00 01 39 :34
5298  86 07 17 05 8B 39 17 04 :88
52A0  CB 0D 0A 3E 3E 20 5A 61 :39
52A8  70 46 4D 0D 0A 20 20 20 :7A
52B0  20 20 20 20 20 20 20 20 :00
52B8  0D 0A 20 20 20 20 52 20 :09
52C0  74 72 61 63 6B 20 73 65 :0D
52C8  63 74 20 20 20 20 20 20 :97
52D0  20 20 20 20 20 72 65 61 :D8
52D8  64 20 31 20 73 65 63 74 :84
52E0  20 61 6E 64 20 64 69 73 :B3
52E8  70 6C 61 79 0D 0A 20 20 :0D
52F0  20 20 57 20 20 20 20 20 :37
52F8  20 20 20 20 20 20 20 20 :00
5300  20 20 20 20 20 20 20 20 :00
5308  20 77 72 69 74 65 20 31 :9C
5310  20 73 65 63 74 0D 0A 20 :06
5318  20 20 20 5A 20 6F 66 66 :15
5320  73 65 74 20 20 20 20 20 :EC
5328  20 20 20 20 20 20 20 20 :00
5330  20 20 63 68 61 6E 67 65 :A6
5338  20 6D 65 6D 6F 72 69 65 :0E
5340  73 0D 0A 20 20 20 20 20 :2A
5348  20 20 20 20 20 20 68 65 :8D
5350  78 2D 62 79 74 65 3C 72 :07
5358  65 74 75 72 6E 3E 20 20 :AC
5360  20 20 20 20 63 68 61 6E :1A
5368  67 65 20 61 6E 64 20 6E :AD
5370  65 78 74 0D 0A 20 20 20 :C8
5378  20 20 20 20 20 20 20 20 :00
5380  27 61 73 63 69 69 2D 63 :C0
5388  68 61 72 3C 72 65 74 75 :37
5390  72 6E 3E 20 20 20 61 73 :52
5398  63 69 69 20 69 6E 0D 0A :43
53A0  20 20 20 20 20 20 20 20 :00
53A8  20 20 20 3C 72 65 74 75 :5C
53B0  72 6E 3E 20 20 20 20 20 :BE
53B8  20 20 20 20 20 20 20 20 :00
53C0  20 6E 6F 20 63 68 61 6E :B7
53C8  67 65 20 61 6E 64 20 6E :AD
53D0  65 78 74 0D 0A 20 20 20 :C8
```

```
53D8  20 20 20 20 20 20 20 20 :00
53E0  69 6C 6C 65 67 61 6C 2D :07
53E8  63 68 61 72 3C 72 65 74 :25
53F0  75 72 6E 3E 20 20 65 78 :B0
53F8  69 74 20 63 6F 6D 6D 61 :0A
5400  6E 64 20 6D 6F 64 65 0D :A4
5408  0A 20 20 20 20 51 20 20 :1B
5410  20 20 20 20 20 20 20 20 :00
5418  20 20 20 20 20 20 20 20 :00
5420  20 20 20 20 71 75 69 74 :43
5428  20 70 72 6F 67 72 61 6D :18
5430  0D 0A 20 20 20 20 48 20 :FF
5438  20 20 20 20 20 20 20 20 :00
5440  20 20 20 20 20 20 20 20 :00
5448  20 20 20 20 20 68 65 6C :D9
5450  70 20 63 6F 6D 6D 61 6E :0B
5458  64 0D 0A 20 20 20 20 50 :4B
5460  20 20 20 20 20 20 20 20 :00
5468  20 20 20 20 20 20 20 20 :00
5470  20 20 20 20 20 20 6F 70 :9F
5478  65 6E 2F 63 6C 6F 73 65 :18
5480  20 70 72 69 6E 74 65 72 :24
5488  0D 0A 0D 0A 00 39 17 02 :80
5490  DB 0D 57 68 69 63 68 20 :FB
5498  44 69 73 6B 20 44 72 69 :CA
54A0  76 65 3A 20 00 17 02 F0 :3E
54A8  17 02 F9 81 00 27 1B 81 :56
54B0  31 26 07 86 01 B7 56 60 :52
54B8  20 10 81 30 26 05 7F 56 :E1
54C0  60 20 07 86 07 17 03 60 :8E
54C8  20 C4 39 17 02 9E 0D 0A :EB
54D0  20 20 20 54 72 61 63 6B :55
54D8  20 00 B6 56 61 17 03 41 :E8
54E0  17 02 89 20 20 53 65 63 :FD
54E8  74 65 72 20 00 B6 56 62 :D9
54F0  17 03 2E 7D 56 63 26 0E :B2
54F8  17 02 71 20 20 53 69 64 :EA
5500  65 20 30 00 20 0C 17 02 :FA
5508  63 20 20 53 69 64 65 20 :48
5510  31 00 39 34 36 17 02 54 :41
5518  0D 0A 20 61 64 72 20 20 :AE
5520  20 20 2B 30 20 2B 31 20 :37
5528  2B 32 20 2B 33 20 2B 34 :5A
5530  20 2B 35 20 2B 36 20 2B :4C
5538  37 20 2B 38 20 2B 39 20 :5E
5540  2B 41 20 2B 42 20 2B 43 :87
5548  20 2B 44 20 2B 45 20 2B :6A
5550  46 0D 0A 2D 2D 2D 2D 2D :3E
5558  2D 2D 2D 2D 2D 2D 2D 2D :68
5560  2D 2D 2D 2D 2D 2D 2D 2D :68
5568  2D 2D 2D 2D 2D 2D 2D 2D :68
5570  2D 2D 2D 2D 2D 2D 2D 2D :68
5578  2D 2D 2D 2D 2D 2D 2D 2D :68
5580  2D 2D 2D 2D 2D 2D 2D 2D :68
5588  2D 2D 2D 00 4F 5F 10 BF :04
5590  55 DA 17 00 5C FD 55 D8 :CC
5598  C6 10 10 8E 55 DE A6 84 :D1
55A0  17 02 7E 81 7F 24 04 81 :40
55A8  20 24 02 86 2E A7 A0 86 :C7
55B0  20 C1 09 26 04 86 7C A7 :BD
55B8  A0 17 02 6C 30 01 5A 26 :D6
55C0  DD 17 01 BB 55 DC FC 55 :32
55C8  D8 C3 00 10 10 83 01 00 :3F
55D0  2D C0 17 02 57 35 36 39 :01
55D8  00 F0 57 6B 20 7C 2E 2E :AA
55E0  2E 2E 2E 2E 2E 2E 7C 2E :BE
55E8  2E 2E 2E 2E 2E 2E 2E 96 :D8
55F0  00 17 02 38 17 02 23 17 :A4
55F8  01 72 20 20 3A 20 00 39 :46
5600  8E 56 64 B6 56 60 A7 07 :62
5608  86 08 A7 84 6F 01 AD 9F :75
5610  FB FA 39 8E 56 64 B6 56 :82
5618  60 A7 07 B6 56 61 A7 04 :26
5620  B6 56 62 A7 05 B6 56 63 :89
5628  A7 06 34 04 CC 56 6C ED :60
5630  02 35 04 86 0A A7 84 6F :65
5638  01 AD 9F FB FA 39 8E 56 :5F
```

```
5640  64 B6 56 60 A7 07 B6 56 :8A
5648  61 A7 04 B6 56 62 A7 05 :26
5650  B6 56 63 A7 06 86 09 A7 :52
5658  84 6F 01 AD 9F FB FA 39 :6E
5660  00 01 01 00 0A 00 56 6C :CE
5668  01 01 00 00 00 FF FF FF :FF
5670  FF 01 C7 03 10 C0 C1 C1 :1C
5678  C0 C0 C0 C0 C6 4D 0E 0F :30
5680  C1 C0 C5 C0 14 15 16 C7 :0C
5688  C0 C0 C0 C0 C0 C0 C0 C0 :00
5690  C0 21 22 33 C6 25 26 27 :6E
5698  28 29 C6 C1 3B 2D 2E 2F :9D
56A0  30 31 32 C3 66 C3 36 37 :EC
56A8  38 39 3A C7 3C 3D 3E C2 :EB
56B0  C2 C0 C2 C0 C3 C0 C2 C0 :09
56B8  C2 C0 4A 4B C4 C1 C0 C5 :21
56C0  C0 C0 C5 C0 C0 C7 C5 C0 :11
56C8  C0 FF FF FF FF FF FF FF :B9
56D0  C0 61 62 63 64 C7 C2 67 :3A
56D8  68 69 6A C5 6C C3 6E 6F :0C
56E0  C2 C1 C1 C5 74 75 C3 C7 :7C
56E8  FF FF FF FF FF FF FF FF :F8
56F0  FF FF FF FF FF FF FF FF :F8
56F8  FF FF FF FF FF FF FF FF :F8
5700  FF FF FF FF FF FF FF FF :F8
5708  C7 FF FF FF FF FF FF FF :C0
5710  FF FF FF FF FF FF FF FF :F8
5718  FF FF FF FF FF FF FF FF :F8
5720  FF FF FF FF FF FF FF FF :F8
5728  FF FF FF FF FF FF FF FF :F8
5730  FF FF FF FF FF FF FF FF :F8
5738  FF FF FF FF FF FF FF FF :F8
5740  FF FF FF FF FF FF FF FF :F8
5748  FF FF FF FF FF FF FF FF :F8
5750  FF FF FF FF FF FF FF FF :F8
5758  FF FF FF FF FF FF FF FF :F8
5760  FF FF FF FF FF FF FF FF :F8
5768  FF FF FF FF 34 12 AE 63 :53
5770  A6 80 81 00 27 05 17 00 :EA
5778  AF 20 F5 AF 63 35 92 34 :D1
5780  32 10 AE 65 AE A4 31 22 :FA
5788  10 AF 65 A6 80 81 00 27 :F2
5790  05 17 00 94 20 F5 35 B2 :AC
5798  0F BF 34 67 BD D8 07 35 :3A
57A0  67 30 01 39 A6 80 81 00 :78
57A8  27 08 81 20 26 04 A1 84 :1F
57B0  27 F2 81 30 25 14 81 39 :BD
57B8  23 13 81 41 25 0C 81 5A :04
57C0  23 0B 81 61 25 04 81 7A :34
57C8  23 03 1A 01 39 1C FE 39 :CD
57D0  7F 57 F7 7F 57 F6 BD 57 :AD
57D8  A4 25 23 8D 27 25 19 34 :12
57E0  04 C6 04 78 57 F7 79 57 :64
57E8  F6 5A 26 F7 35 04 BB 57 :B8
57F0  F7 B7 57 F7 20 E0 00 00 :FC
57F8  BD 57 A4 24 FB 39 FC 57 :63
5800  F6 1C FE 39 80 47 2A 0F :49
5808  8B 06 2A 04 8B 07 2A 07 :82
5810  8B 0A 2B 03 1C FE 39 1A :30
5818  01 39 34 02 BD AC 37 35 :45
5820  82 34 02 BD AC 3D 35 82 :15
5828  BD D0 8E 39 34 67 BD 9B :47
5830  50 35 E7 86 50 97 C3 86 :22
5838  19 B7 03 0D 4F B7 03 0E :F7
5840  86 18 B7 03 0F 34 77 BD :CF
5848  C8 BC 35 F7 7D 05 AC 26 :04
5850  2C 17 FF 18 20 20 20 2A :E4
5858  20 4F 75 74 70 75 74 20 :D1
5860  61 6C 73 6F 20 74 6F 20 :D2
5868  74 68 65 20 70 72 69 6E :1A
5870  74 65 72 20 2A 0D 0A 00 :AC
5878  73 05 AC 20 25 7F 05 AC :99
5880  17 FE E9 20 20 20 2A 20 :A8
5888  50 72 69 6E 74 65 72 20 :04
5890  6F 75 74 70 75 74 20 63 :34
5898  6C 6F 73 65 64 20 2A 0D :6E
58A0  0A 00 39 00 00 00 00 00 :43
```

特集 ディスク活用術

ディスクユーティリティ集

for FM-7
FLEX→F-BASICファイルコンバータ

Simple Logic

F-BASIC上で走る使い勝手のよいエディタ（プログラムのソーステキストを書くツール）やアセンブラ（ニモニックを機械語に翻訳するツール）がない。そこで，機械語プログラムが欲しい人には，FLEXの上で開発し，FLEXのバイナリファイルとして作ってから，FLEXからF-BASICへのファイルコンバータによってそれをF-BASICのバイナリファイルに直す，という人が多い。

このプログラムは，そんな人が愛用しているであろうツールを，使いやすさと機能と速さで少しだけリファインしたものです。

仕　様

入力ファイルはFLEXのバイナリファイルとテキストファイルで，ファイル属性がBIN，CMD，TXTのもの。ただしバイナリファイルは，そのモジュールの数が10個以下，大きさが約10Kバイト以下のものという制限がある。これ以上のモジュール数，あるいは大きさをもつファイルについては，そのファイルを分割してコピーすることをおすすめする。テキストファイルについては，とくに制限なし。

出力ファイルはF-BASICのバイナリ，およびテキストファイル。ディスクファイルについては，ドライブ 0に収まっているディスケット中のファイル，すなわちファイルディスクリプタのデバイスネームが0：のもののみと制限。構造のないファイル，SCRN：とLPT0：にも出力可能だが，これはテキストファイルに限る。

使い方

使い方はいたって簡単。入力ファイルを読んでF-BASICのファイルへ出力するというのが仕事。F-BASICのディスケットを0に，FLEXのディスケットを1に入れ，入力ファイル名と出力ファイル名を教えてやるだけでOK。

ただ，お金がない人はいつも身体を使わないといけなくなるのだが，このプログラムでもそう。ディスクドライブを1台しか持ってない人は，ディスクファイル間コピーをするときには，モニタ上の指令に従ってディスケットの入れ替えをやらないといけない。でも，汗が出るほどではないからね。

入出力ファイル名を入れてやると，それに従ってコピーを行って，コピー終了後，次に行うコピーの入出力ファイル名を聞いてくるから，続けるときはファイル名を教えてあげる。やめるときはBreakキーを押す，という乱暴な幕切れが必要なプログラムです。

入力ファイル名の書き方

入力ファイルはFLEXのファイル。FLEXのファイル名の書き方そのままです。ドライブが1台しかない人は，FLEXのディスクもF-BASICのディスクもドライブ0に入れることになるので，ファイル名の最初のデバイスナンバを0.にすること。

エクステンション（ファイル属性）は，BINとCMDとTXTだけ受け付ける。BINとCMDはバイナルファイルとして，TXTはテキストファイルとして扱われ，それぞれF-BASICのファイルへコピーを入れる。

この入力ファイル名の入力のときに，ただリターンキーを押すと，FLEXのディスクのカタログが表示されるので，何をコピーするのか忘れてしまったときに便利です。

出力ファイル名の書き方

これもF-BASICのファイルディスクリプタの書き方そのまま。ディスクのファイルなら，デバイスネームは不要。画面とプリンタにテキストファイルを出力するときだけ，デバイスネーム〝SCRN:と〝LPT0:をお忘れなく。このとき，文字列を表すダブルコーテーションが，コロンと込みの場合には絶対必要ですよ。

走らせる

リスト1のファイル名はMAIN，**リスト2**はTRANSとしてセーブしてください。

RUN "MAIN"↵

で，機械語部分〝TRANS〟がロードされていなければこれをロードし，されていればそのままスタート。あとは画面があなたを導いてくれるはずです。

おわりに

今回のバージョンはFM-7用のものですが，次号までにはFM-8でも走らせる予定です。

コピーツールがあっても，FLEXのプログラムがそのままF-BASICで走るわけではないのです，当たり前だけど。FLEX上のプログラムは，全て入出力関係を中心にFLEXのシステムサービス（備え付けのルーチン）に頼っているのに，F-BASICの上では，そのFLEXのシステムルーチンがないわけで，当然プログラムは暴走してしまう。

たとえば，このコピープログラムも初め

はオールBASICで作り，一部機械語に直すという手順をとって，その機械語部分をFLEXで開発したのですが，そこに現れる文字列出力ルーチンは，FLEXのシステムに用意されているものは使用できず，BASICのROM上のルーチンをコールしました。

よって，F-BASIC上で走る機械語プログラムをFLEXで開発せんとする者は，まずBASIC用の入出力関係の下位ルーチンから作っていかなければいけないというわけなのです。

リスト1 MAIN

©1983 Simple Logic

```
100 '
110 '    FLEX to F-BASIC file converter program
120 '                                on F-BASIC
130 '                   written by G.HIRANO
140 '                          on 9-dec-1984
150 '
200 CLEAR 1000,&H3FFF
300 IF PEEK(&H4100)<>&H34 OR PEEK(&H44F1)<>&H34 OR PEEK(&H43EF)<>&H34 THEN
    LOADM"TRANS
1000 GOSUB 2000 'initialize
1010 WHILE 1
1020    GOSUB 3000 'input source & distination file names
1030    GOSUB 4000 'seek the file and the first secter
1040    IF NOT FIND THEN BEEP :PRINT"  File could not be found
1050    IF FIND THEN   GOSUB 5000 'load & check(& save)
1060    IF FIND AND EXT=0 THEN   GOSUB 6000 'save files
1070    IF NOT FIND AND EXT=1 THEN CLOSE LGNO
1090 WEND
1100 END
1200 '
1990 '   initialize
1995 '
2000 DEF USR0=PEEK(&HFBFA)*256+PEEK(&HFBFB)
2002 DEF USR1=&H44F1
2004 DEF USR2=&H4100
2006 DEF USR3=&H43EF
2010 WIDTH80,25
2015 DIM LDADR(10),EDADR(10)
2017 FIELD #0,255 AS DIR$
2020 RETURN
2899 '
2900 '   get file names
2995 '
3000 PRINT
3010 INPUT" Input Source File Name (Upper Case):",SOURCE$
3020    P=INSTR(SOURCE$,".")
3030    IF P=0 OR P=1 THEN GOSUB 3300:GOTO 3010
3035    SOURCENAME$=LEFT$(SOURCE$,P-1):UNIT=1
3036    IF SOURCENAME$="0" THEN UNIT=0:P1=INSTR(3,SOURCE$,"."):GOSUB3200
3037    IF SOURCENAME$="1" THEN UNIT=1:P1=INSTR(3,SOURCE$,"."):GOSUB3200
3050    SOURCEEXT$=MID$(SOURCE$,P+1,3)
3060    EXT=-1
3070    IF SOURCEEXT$="CMD" OR SOURCEEXT$="BIN"  THEN EXT=0
3080    IF SOURCEEXT$="TXT"  THEN EXT=1
3090    IF EXT<>0 AND  EXT<>1 THEN GOTO 3010
3100 INPUT" Input Distination File Name :",DISTIN$
3102    IF INSTR(DISTIN$,CHR$(&H22))<>0 OR INSTR(DISTIN$,"(")<>0 OR
        INSTR(DISTIN$,")")<>0 THEN GOTO 3100
3104    IF INSTR(DISTIN$,":")=0 THEN GOTO 3110
3105    IF LEFT$(DISTIN$,2)="0:" THEN DISTIN$=MID$(DISTIN$,3,8):GOTO 3110
3106    IF EXT=1 AND (LEFT$(DISTIN$,5)="SCRN:" OR LEFT$(DISTIN$,5)="LPT0:")
        THEN LGNO=2:OPEN "O",#2,DISTIN$:GOTO3134
3108    GOTO 3100
3110    IF LEN(DISTIN$)>7 OR DISTIN$="" THEN GOTO 3100
3120    IF EXT=1 THEN LGNO=1:GOSUB 4530:OPEN "O",#1,DISTIN$:GOTO 3134
3125    IF UNIT=0 THEN GOSUB 4530
3130    SAVEM DISTIN$+"0",&H1000,&H1010,&H1000:KILL DISTIN$+"0"
3134 RETURN
3200 SOURCENAME$=MID$(SOURCE$,P+1,P1-P-1):P=P1:RETURN
3300 DUMMY=USR3(0):RETURN
3843 (0):RETURN
3899 '
3900 '   search the file
```

```
3995 '
4000 IF UNIT=0 AND LGNO=1 THEN GOSUB 4500
4005 TARGET$=LEFT$(SOURCENAME$+STRING$(8,CHR$(0)),8)+SOURCEEXT$
4010 DIR$=STRING$(240,"a"):DUM1=PEEK(VARPTR(DIR$)+1):DUM2=PEEK(VARPTR(DIR$)+2)
4015 POKE VARPTR(DIR$)+1,&H40:POKE VARPTR(DIR$)+2,16
4020 SIDE=0:TRACK=0:SECTER=5:ADR=&H4000:GOSUB 4610
4030 FIND=0:SEARCH=-1
4040 WHILE SEARCH
4050   PSOURCE=INSTR(1,DIR$,TARGET$)
4060   IF PSOURCE=0 THEN GOTO 4100
4090   SEARCH=0:FIND=-1:TRACK=ASC(MID$(DIR$,PSOURCE+13,1))
       :SECTER=ASC(MID$(DIR$,PSOURCE+14,1)):GOTO 4200
4100   IF ASC(MID$(DIR$,217,1))=0 THEN SEARCH=0:FIND=0:GOTO 4200
4150   TRACK=PEEK(ADR):SECTER=PEEK(ADR+1): GOSUB 4600
4200 WEND
4205 POKE VARPTR(DIR$)+1,DUM1:POKE VARPTR(DIR$)+2,DUM2
4210 RETURN
4500 IF UNIT=1 THEN GOTO 4520
4505 PRINT"  Insert FLEX Diskette to Drive 0"
4510 INPUT"  then Hit RETURN Key  ",A$
4520 RETURN
4530 IF UNIT=1 THEN GOTO 4550
4535 PRINT"  Insert F-BASIC Diskette to Drive 0"
4540 INPUT"  then Hit RETURN Key  ",A$
4550 RETURN
4600 IF SECTER>16 THEN SIDE=1 ELSE SIDE=0
4610 D#=1:RCB=VARPTR(D#)
4615 IF SECTER<1 OR SECTER>32 OR TRACK>39 OR TRACK<0 THEN GOTO 7000
4620 POKE RCB,10
4630 POKE RCB+1,0
4640 POKE RCB+2,ADR¥256
4650 POKE RCB+3,ADR MOD 256
4660 POKE RCB+4,TRACK
4670 POKE RCB+5,SECTER
4680 POKE RCB+6,SIDE
4690 POKE RCB+7,UNIT
4700 DUMMY=USR0(D#)
4710 IF PEEK(RCB+1)<>0 THEN GOTO 7000
4720 RETURN
4899 '
4900 '   load the file (and save if txt)
4995 '
5000 IF EXT=1 THEN GOTO 5500
5010 '
5020 D#=1:P=VARPTR(D#)
5030 POKE P+2,TRACK
5040 POKE P+3,SECTER
5050 POKE P+4,UNIT
5060 DUMMY=USR1(D#)
5070 IF PEEK(P+1)<>0 THEN ANS=PEEK(P+3):ERROR ANS
5080 PLDADR=&H44BA
5090 LSTART=&H46B7
5100 RETURN
5110 '
5500 D#=1:P=VARPTR(D#)
5510 POKE P+2,TRACK
5520 POKE P+3,SECTER
5530 POKE P+4,UNIT
5540 POKE P+5,LGNO
5550 DUMMY=USR2(D#)
5560 IF PEEK(P+1)<>0 THEN ANS=PEEK(P+3):ERROR ANS
5570 CLOSELGNO
5580 RETURN
5899 '
5900 '   save the binery file
5995 '
6000 GOSUB 6100
6005 OFSET=0:IF UNIT=0 THEN GOSUB 4530
6010 FOR I=0 TO MDNO
6015   PRINT:PRINT "  > ";LEFT$(DISTIN$+RIGHT$(STR$(I),1)+"        ",8);
6017  PRINT "  ";HEX$(LDADR(I));" -> ";HEX$(EDADR(I));"  jump to ";HEX$(TRAD)
```

```
6020   SAVEM DISTIN$+RIGHT$(STR$(I),1),LSTART+OFSET,LSTART+OFSET+EDADR(I)-LDADR(
I),TRAD
6030   OFSET=OFSET+EDADR(I)-LDADR(I)+1
6040 NEXT I
6050 GOSUB 6200
6060 RETURN
6100 MDNO=PEEK(PLDADR+41)/2
6110 FOR I=0 TO MDNO
6120 LDADR(I)=PEEK(PLDADR+I*2)*256+PEEK(PLDADR+I*2+1)
6130 EDADR(I)=PEEK(PLDADR+I*2+20)*256+PEEK(PLDADR+I*2+21)
6140 NEXT
6150 TRAD=256+PEEK(PLDADR+42)*256+PEEK(PLDADR+43)
6160 RETURN
6200 FOR I=0 TO MDNO
6210   GOSUB 6400
6220   ADR=&H4000
6230   UNIT=0
6250   GOSUB 4610
6330   POKE ADR+3,PEEK(PLDADR+I*2)
6340   POKE ADR+4,PEEK(PLDADR+I*2+1)
6350   POKE RCB,9
6360   DUMMY=USR0(D#)
6370 NEXT
6380 RETURN
6400 '
6410 FOR SECTER=4 TO 32
6420   D$=DSKI$(0,1,SECTER)
6430   P=INSTR(D$,DISTIN$+RIGHT$(STR$(I),1))
6440   IF P=0 THEN GOTO 6460
6450   IF MID$(D$,P+12,1)=CHR$(0) THEN SECTER=32:CLUS=ASC(MID$(D$,P+14,1))
6460 NEXT
6470 TRACK=CLUS¥4+2
6480 SECTER=(CLUS MOD 2)*8+1
6490 SIDE=(CLUS MOD 4)¥2
6500 RETURN
6899 '
6900 '    error occured
6995 '
7000 '
7010 PRINTERL:CLOSE LOGNO
7020 ERROR ERR
7995 '
```

リスト2 TRANS

©1983 Simple Logic

Add	+0	+1	+2	+3	+4	+5	+6	+7	Sum
4100	34	10	A6	02	B7	46	24	A6	:B3
4108	03	B7	46	23	A6	04	B7	46	:CA
4110	22	A6	05	B7	43	06	81	01	:4F
4118	10	27	00	C1	81	02	10	27	:B2
4120	00	08	86	05	B7	44	E8	16	:8C
4128	04	41	17	04	7D	10	25	04	:16
4130	3A	17	01	B8	7D	46	23	10	:00
4138	27	00	11	17	04	9B	10	25	:23
4140	04	29	17	00	25	10	25	04	:A2
4148	22	16	FF	E8	BE	45	F7	8C	:A5
4150	00	FF	10	24	00	03	16	FF	:4B
4158	E2	17	04	8F	17	00	0B	10	:BE
4160	25	04	08	35	10	6F	01	1C	:02
4168	FE	39	81	0D	10	27	00	18	:14
4170	81	09	10	27	00	1D	81	00	:5F
4178	10	27	00	09	81	18	10	27	:10
4180	00	03	BD	D0	8E	1C	FE	39	:71
4188	BD	D0	8E	86	0A	BD	D0	8E	:C6
4190	16	FF	F2	17	00	17	10	25	:6A
4198	00	3B	81	01	2F	37	1F	89	:CB
41A0	86	20	BD	D0	8E	5A	10	26	:51
41A8	FF	F8	16	FF	D8	86	01	B1	:1C
41B0	43	06	10	27	00	04	17	04	:9F
41B8	20	39	10	BC	44	E6	10	24	:83
41C0	00	05	A6	A0	1C	FE	39	17	:B5
41C8	00	A0	17	04	0C	34	02	17	:14
41D0	00	98	35	02	39	86	47	B7	:8C
41D8	44	E8	1A	01	39	CC	46	B7	:49
41E0	FD	44	E6	7F	42	69	17	03	:6B
41E8	C1	10	25	03	7E	17	00	28	:B6
41F0	10	25	03	77	17	00	73	17	:50
41F8	00	56	10	25	03	6D	7D	46	:BE
4200	23	10	26	00	0D	BE	45	F7	:60
4208	8C	00	FF	10	23	00	03	16	:D7
4210	FF	51	17	00	55	16	FF	D5	:A6
4218	10	8E	46	B7	7D	46	23	10	:91
4220	27	00	14	17	03	B3	10	25	:3D
4228	00	25	A7	A0	10	8C	70	00	:78
4230	10	24	00	15	16	FF	E5	BE	:01
4238	45	F7	8C	00	FF	10	24	00	:FB
4240	03	16	FF	DF	17	03	A4	A7	:5C
4248	A0	10	BF	44	E6	1C	FE	39	:EC
4250	10	8E	46	B7	17	00	95	A6	:ED
4258	A0	17	FF	0E	10	25	00	08	:01
4260	10	BC	44	E6	10	25	FF	EF	:19
4268	39	00	7D	46	22	10	26	00	:54
4270	68	17	00	93	0D	0A	20	20	:69

```
4278  49 6E 73 65 72 74 20 00 :95
4280  7D 42 69 10 27 00 37 17 :AD
4288  00 7D 46 4C 45 58 20 44 :10
4290  69 73 6B 65 74 74 65 20 :19
4298  20 0D 0A 00 17 00 68 20 :D6
42A0  20 20 54 68 65 6E 20 68 :57
42A8  69 74 20 52 45 54 55 52 :8F
42B0  4E 20 4B 65 79 20 00 17 :CE
42B8  00 79 73 42 69 39 17 00 :E7
42C0  46 46 2D 42 41 53 49 43 :1B
42C8  20 44 69 73 6B 65 74 74 :F8
42D0  65 20 20 0D 0A 00 16 FF :D1
42D8  C3 7D 42 69 10 27 00 06 :28
42E0  17 00 1F 16 FF D4 17 00 :36
42E8  0D 16 FF CE 34 04 F6 43 :61
42F0  06 D7 BF 35 04 39 7F 46 :D3
42F8  22 17 03 29 86 01 B7 46 :E9
4300  22 39 17 03 20 39 00 34 :02
4308  12 AE 63 A6 80 81 00 27 :F1
4310  05 17 00 B3 20 F5 AF 63 :F6
4318  35 92 34 32 10 AE 65 AE :FE
4320  A4 31 22 10 AF 65 A6 80 :41
4328  81 00 27 05 17 00 98 20 :7C
4330  F5 35 B2 0F BF 34 67 BD :02
4338  D8 07 35 67 30 01 39 A6 :8B
4340  80 81 00 27 08 81 20 26 :F7
4348  04 A1 84 27 F2 81 30 25 :18
4350  14 81 39 23 13 81 41 25 :EB
4358  0C 81 5A 23 0B 81 61 25 :1C
4360  04 81 7A 23 03 1A 01 39 :79
4368  1C FE 39 7F 43 92 7F 43 :69
4370  91 BD 43 3F 25 23 8D 27 :CC
4378  25 19 34 04 C6 04 78 43 :FB
4380  92 79 43 91 5A 26 F7 35 :8B
4388  04 BB 43 92 B7 43 92 20 :40
4390  E0 00 00 BD 43 3F 24 FB :3E
4398  39 FC 43 91 1C FE 39 80 :DC
43A0  47 2A 0F 8B 06 2A 04 8B :CA
43A8  07 2A 07 8B 0A 2B 03 1C :17
43B0  FE 39 1A 01 39 34 02 0F :D0
43B8  BF BD AC 37 35 82 34 02 :4C
43C0  0F BF BD AC 3D 35 82 0F :3A
43C8  BF BD D0 8E 39 0F BF 34 :15
43D0  67 BD 9B 50 35 E7 86 50 :01
43D8  97 C3 86 19 B7 03 0D 4F :0F
43E0  B7 03 0E 86 18 B7 03 0F :2F
43E8  34 77 BD C8 BC 35 F7 34 :4C
43F0  10 86 01 B7 46 22 17 00 :CD
43F8  1B 10 24 00 0A 7F 46 22 :40
4400  17 00 11 10 25 00 05 35 :97
4408  10 1C FE 39 86 48 B7 44 :2C
4410  E8 16 01 57 17 FE F0 0D :68
4418  0A 20 20 20 43 61 74 61 :E3
4420  6C 6F 67 20 6F 66 20 46 :9D
4428  4C 45 58 20 64 69 73 6B :B4
4430  20 0D 0A 00 86 00 B7 46 :BA
4438  24 86 05 B7 46 23 17 01 :E7
4440  69 10 25 00 24 30 89 40 :BB
4448  0C C6 0A 34 04 17 00 1A :45
4450  35 04 10 27 00 0B 5A 10 :E5
4458  26 FF F0 7C 46 23 16 FF :0F
4460  DD 17 FE A3 0D 0A 00 1C :C8
4468  FE 39 5F 10 8E 44 A1 A6 :BF
4470  85 10 27 00 44 81 FF 10 :90
4478  27 00 39 A7 A5 5C A6 85 :33
4480  26 02 86 20 A7 A5 5C C1 :37
4488  08 10 25 FF F1 31 21 A6 :25
4490  85 26 02 86 20 A7 A5 5C :FB
4498  C1 0B 25 F3 17 FE 68 20 :81
44A0  20 2A 2A 2A 2A 2A 2A 2A :46
44A8  2A 2E 2A 2A 2A 20 20 20 :36
44B0  20 3A 20 00 30 88 18 1C :66
44B8  FE 39 00 00 00 00 00 00 :37
44C0  00 00 00 00 00 00 00 00 :00
44C8  00 00 00 00 00 00 00 00 :00
44D0  00 00 00 00 00 00 00 00 :00
44D8  00 00 00 00 00 00 00 00 :00
44E0  00 00 00 00 00 00 00 00 :00
44E8  00 00 00 00 00 00 00 00 :00
44F0  00 34 10 A6 02 B7 46 24 :0D
44F8  A6 03 B7 46 23 A6 04 B7 :2A
4500  46 22 4F 5F FD 44 BA FD :0E
4508  44 CE FD 44 E2 CC 46 B7 :FE
4510  FD 44 E6 17 00 94 10 25 :07
4518  00 51 17 00 BC 10 25 00 :59
4520  4A 81 02 27 11 81 16 27 :C3
4528  1D 7D 46 23 27 2E 86 47 :25
4530  B7 44 E8 16 00 35 17 00 :45
4538  FF 10 25 00 2E 17 00 3B :B4
4540  10 25 00 27 20 14 17 00 :A7
4548  90 10 25 00 1E B7 44 E4 :C2
4550  17 00 86 10 25 00 14 B7 :9D
4558  44 E5 20 BE 35 10 6F 01 :BC
4560  CC 46 B7 ED 02 CC 44 BA :82
4568  ED 04 39 35 10 1C FE 86 :0F
4570  FF A7 01 6F 02 B6 44 E8 :FA
4578  A7 03 39 BE 44 E6 7D 45 :8D
4580  A9 27 20 34 10 17 00 51 :9C
4588  35 10 10 25 00 11 A7 80 :B2
4590  8C 70 00 24 05 7A 45 A9 :8D
4598  20 E4 86 07 B7 44 E8 1A :8E
45A0  01 20 05 BF 44 E6 1C FE :29
45A8  39 00 17 00 4C 10 25 00 :D1
45B0  20 B6 40 00 B7 46 24 4D :84
45B8  2B 17 81 27 22 13 B6 40 :15
45C0  01 B7 46 23 81 20 22 09 :ED
45C8  8E 00 04 BF 45 F7 1C FE :A7
45D0  39 86 47 B7 44 E8 1A 01 :04
45D8  39 BE 45 F7 8C 00 FF 23 :E1
45E0  0A 17 FF C6 10 25 00 0E :29
45E8  BE 45 F7 A6 89 40 00 30 :99
45F0  01 BF 45 F7 1C FE 39 00 :4F
45F8  00 8E 44 E9 86 0A A7 84 :76
4600  6F 01 CC 40 00 ED 02 B6 :21
4608  46 24 A7 04 B6 46 23 A7 :DB
4610  05 4A 44 44 44 44 A7 06 :0C
4618  B6 46 22 A7 07 AD 9F FB :13
4620  FA 39 01 FF FF 8E 44 E9 :ED
4628  B6 46 22 A7 07 86 08 A7 :01
4630  84 6F 01 AD 9F FB FA 39 :6E
4638  17 FF 9E 10 25 00 6B B7 :0B
4640  46 B5 17 FF 94 10 25 00 :DA
4648  61 B7 46 B6 FC 44 BA 26 :34
4650  0A FC 44 CE 26 05 FC 44 :83
4658  E2 27 3B BE 44 E2 EC 89 :9D
4660  44 CE C3 00 01 10 B3 46 :DF
4668  B5 26 1B 17 FF 6B 10 25 :AC
4670  00 38 B7 45 A9 1F 89 4F :D4
4678  BE 44 E2 E3 89 44 CE ED :4F
4680  89 44 CE 16 00 2C FC 44 :1D
4688  E2 C3 00 02 FD 44 E2 83 :4D
4690  00 12 10 2A 00 14 BE 44 :62
4698  E2 FC 46 B5 ED 89 44 BA :4D
46A0  83 00 01 ED 89 44 CE 16 :22
46A8  FF C1 86 50 B7 44 E8 1A :93
46B0  01 39 1C FE 39 00 00 00 :8D
```

特集 ディスク活用術

ディスクユーティリティ集

スキューをかけるプログラム
――知っていると差がつくDISKの知識

鶴岡 哲明

「スキューをかけると速くなる」という話を耳にした人も多いと思います。しかし，スキューというものを理解するには，ディスクのフォーマットについて多少知る必要があります。この辺のことは意外に簡単なのに，知っている人は少ないようです。

そこで，ディスクのフォーマットについて簡単に説明し，スキューとは何か，そしてFMにスキューは効果があるかについて述べてみましょう。

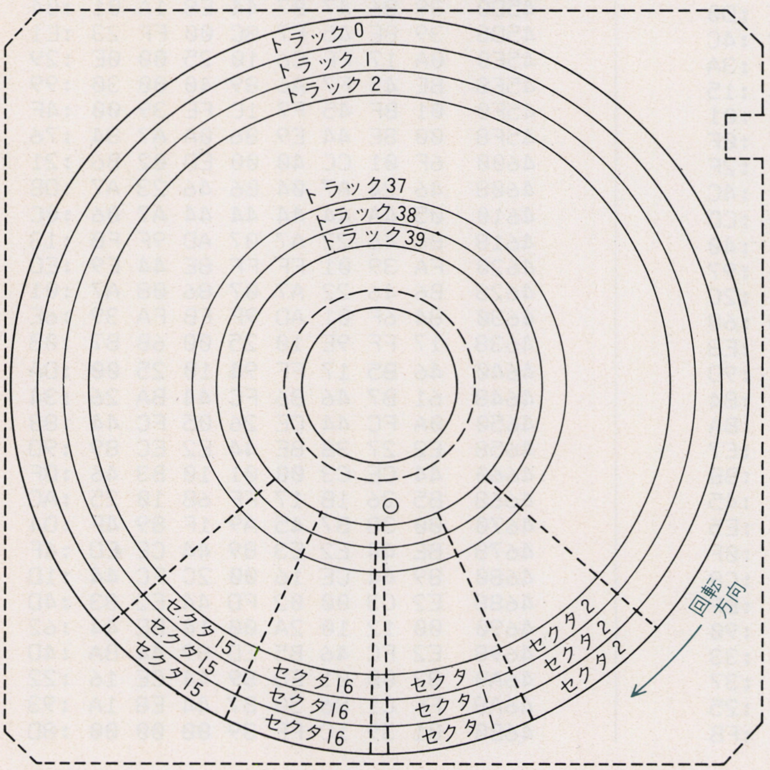
ディスクの読み/書きの動作は？

フォーマットの説明に入る前に，ディスクの読み出し／書き込み時の様子について考えてみましょう。

ディスクはご存知のようにデータを1セクタ単位で読み書きします。しかし，大抵の処理は1セクタのみで終了しません。すると，たとえばロード時には1セクタ読んでは種々の処理をし，さらにディスクからの読み出し，…… を繰り返すことになります。なお，ここで言う種々の処理とは，内部の処理，――次に読み出すセクタを定めたりデータをテキストエリアに転送したりする処理のことです。つまり，ディスクを読み出して，しばらくしてからまた読み出すわけです。ところで，このセクタは図1のように1つのトラックに16個並んでいます。ディスクは1分間に300回転（5インチの場合）しかしないので，目的のセクタにすぐ到達できるわけではありません。最悪でディスク1回転＝0.2秒待つことになります。この時間は何回も積もると，相当なロスタイムになります。

図1

F-BASICのディスク構造



ディスクのフォーマットは？

ところで，ディスクのトラック，セクタの位置は絶対的に決まっているのでしょうか。

答えはトラックについてはYes，セクタについてはNoです。多くの人がこの点を誤解しているようです。ただ，ほとんどのディスクは，インデックスホールの位置から順にセクタ1，2，……，16と並んでいますので，こう誤解してもやむを得ないかもしれません。

しかし，セクタの位置が絶対的に決まっている(例えばインデックスホールから5/16回転した所がセクタ5）わけではないのなら，どうやってセクタの位置を確定させるのでしょうか。

実は1つ1つのトラックに「ここから××番セクタが始まるよ」という信号が記録してあるのです。

もっと正確にいうと,「後ろに1セクタ分のデータがあり，そのデータは第×トラックの第×セクタのデータである」と書かれた場所があり，その後に1セクタ分のデータが並んでいるのです。前者をアドレスフィールド，後者をデータフィールドと呼びます。やさしく言えば，トラック1周にまず第何セクタか示す旗が立っていて，そのすぐ後にデータが並んでいる感じになります(図2)。

そして新品のディスクをフォーマットするというのは，この旗を立てる作業をすることなのです。旗が立っていないと，いくら目的のセクタを捜しても見つからず，

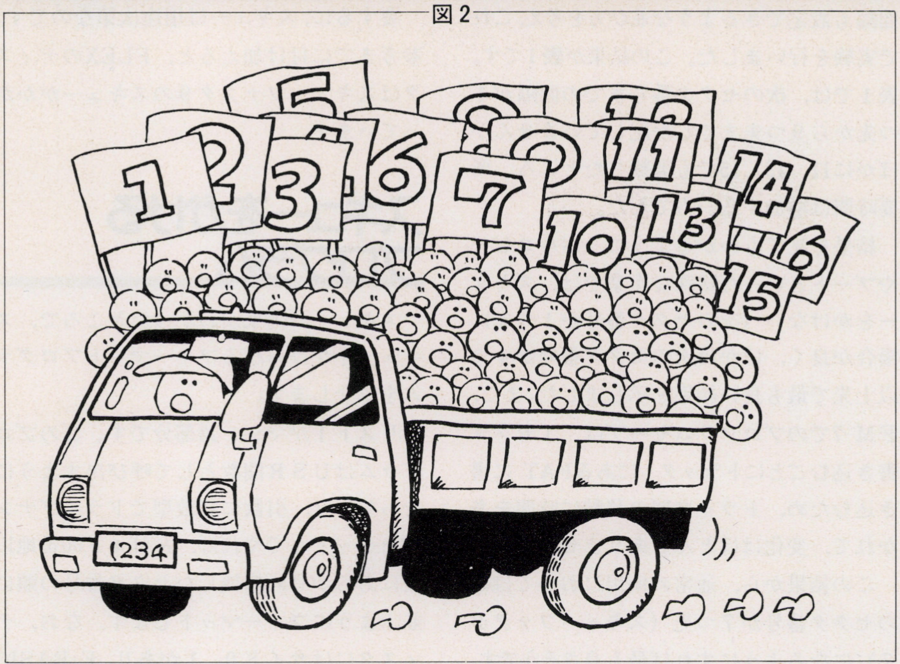
```
Device I/O Error
```

となってしまうわけです。

そして，ディスクの通常の書き込みは，データの部分だけ書き換えるだけなのです。

さてこの旗ですが，1つのトラックのど

図 2



こに立っていてもよいかというと，そういうわけなのです。セクタ1の旗がインデックスホールのすぐ後ろになくてもよいのです。

ここまで話を進めれば何が言いたいか賢明な読者ならおわかりでしょう。

つまり，セクタの順番は決まっているわけではなく，セクタが始まるという印さえちゃんとついていればいいのです。フォーマットの段階で，旗を1，4，7，……，16，3，……などと番号順ではない順で立ててやれば，セクタの位置は図1のようにはならないわけです。参考までにこのようにセクタの位置を決めるやり方をソフトセクタと言い，インデックスホールがセクタ数だけあいていて，これで絶対的な位置を決めるハードセクタもありますが，こちらはパソコンで使用されてる例を私は知りません。

スキューとは？

さて，ここで話を始めに戻します。ディスクの読み出し，書き込みは1セクタずつ行われ，1セクタ読み／書きした後，必要な処理をすると述べました。そして，この処理をしている間にもディスクは回り，次のセクタを読み出すときにはすでにそのセクタは通り過ぎた後であることが多いのです。それは，セクタを複数読み／書きするときは，連続したセクタ番号であるケースがほとんどであり，前述のように通常はセクタが番号順に並んでいるからなのです。

例えば，今，セクタ1を読み込み，次にセクタ2を読み出そうとするとします。しかし，セクタ1を読み終えた後に必要な処理にかかる時間と，セクタ読み出しのコマンド（FDC：フロッピディスクコントローラに送られる）を送ってから読み出し可能となるまでの時間のため，ディスクから読み出せるようになったときには，セクタ5とか6とかの位置になってしまい，求めるセクタ2まであと3/4回転もしなくてはならないのです。この待ち時間を減らす手段がスキューです。

スキューには，いくつか方法があります。1つは実際のセクタ番号（フォーマット時ディスクに規定される）と論理セクタ番号（コンピュータ内で扱われるセクタ番号）を違える方法です。今の例で言えば，実際のセクタ6をセクタ2としてコンピュータに処理させます。セクタ2を読もうとして，セクタ6を読み出すわけで，今度はすぐこのセクタに到達し，待ち時間はなくなります。この実際のセクタ番号と仮想セクタ番号の関係は，グラフィックのカラーコードとパレットコードの関係に似ています。しかし，この方法は，プログラムがこのようにできていないと実現できません。F-BASICではこのようなことはやっていないのです。

もう1つの方法は，ディスク上のセクタ位置を入れ替える方法です。これは前述のセクタ番号の旗の並び方をフォーマット時に変えてやれば実現できます。先述の例では，4～5セクタ先にセクタ2を作ってやればよいわけです。この方法は，フォーマットのときだけ特別のプログラムを用いるだけで済むので，実用的です。

スキューは速いか？

スキューには，スキューファクタ（インタリーブ）という用語がつきものです。スキューファクタとは，次にあるセクタの番号がいくつ先のセクタ番号かを示す数です。たとえば，スキューファクタ1なら通常の

図 3

セクタ番号の並んでいる順 →

スキュー ファクタ	1	:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	3	:	1	4	7	10	13	16	3	6	9	12	15	2	5	8	11	14
	5	:	1	6	11	16	5	10	15	4	9	14	3	8	13	2	7	12
	7	:	1	8	15.	6	13	4	11	2	9	16	7	14	5	12	3	10
	9	:	1	10	3	12	5	14	7	16	9	2	11	4	13	6	15	8
	11	:	1	12	7	2	13	8	3	14	9	4	15	10	5	16	11	6
	13	:	1	14	11	8	5	2	15	12	9	6	3	16	13	10	7	4
	15	:	1	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2

フォーマット，3ならセクタ1，4，7……と並んでいます。セクタ番号が16を超えたら16を引いて続けていくわけです(図3)。スキューファクタ11を例にとると，セクタ2は1の3つ先，セクタ3は2の3つ先…となり，次の番号のセクタにたどりつくのに時間がかかるようになります。このスキューファクタを，先ほどの内部処理（準備終了まで）にかかる時間と合うように設定してやれば，大幅に待ち時間が減り，ディスクの読み／書きが速くなります。

しかし，当然のことながら，この準備終了までの時間が変わると，最適スキューファクタも異なります。そこで，様々な用途に際してスキューファクタを変えたときの所要時間を調べる必要が出てきます。

しかし，普通のスキューでは，次のセクタ番号までの間隔を奇数個のセクタ分でしか設定できないのは，図3を見てのとおりです。そこでひと工夫して偶数個のセクタ間隔も設定できるような並び方を考え，これで実験を行いました。この結果が**表1**です。表1では，次のセクタ番号までの間隔が1つ先から9つ先までしか記していませんが，ほかに11，13，15でも実験したところ，所要時間は増加する一方でした。

結果を見てもわかるように，ボルコピーやブートといった特殊な作業では，スキューをかけない（次のセクタ番号が1つ先）場合が良く，ほかは次のセクタ番号が4つ以上先で最も良い結果を示しました。また，FM-7でのプログラムセーブは，1セクタ書き込むごとにトラック1にあるFAT に書き込むため，トラック間の移動に時間をさかれて，変化はほとんどありません。

この結果から，通常の使用においては次のセクタ番号が7つ先（スキューファクタ7）になるようにすれば最も良さそうです。

また，用途が特定される場合は，表1を参考にスキューをかけてみてください。

要するに，スキューは用途次第なのです。参考までに付け加えると，FLEXのディスクはスキューファクタ9のスキューがかかっています。

スキューをかける プログラム

スキューの効果がわかったところで，スキューをかけられるフォーマットプログラムを紹介します。

リスト1がマシン語部分です。このプログラムはUSR関数として呼び出すようになっており，引数に整数型でドライブナンバをセットしてやれば，&H4F00番地以後にあらかじめ格納したセクタ番号の順に並ぶようにフォーマットします。なお，ディスクにはサイド0，1があり，F-BASICでは，サイド1の1～16セクタを17～32セクタと呼んでいることからもわかるように，

表1　スキュー ベンチマークテスト

次の番号のセクタまでの間隔＼テストNo.	1	2	3	4	5	6	7	8	9	10	11	12	13
1	47	48	13	88	47	24	131	21	21	25	148	21	22
2	62	66	14	93	48	26	132	22	22	26	149	23	24
3	78	81	16	95	20	26	131	23	24	27	150	23	24
4	90	97	17	100	18	27	131	24	7	27	150	25	21
5	106	108	20	35	20	28	131	25	8	29	150	26	15
6	121	124	22	38	22	13	131	8	9	21	150	16	11
7	137	139	23	42	23	12	131	9	10	15	151	11	12
8	152	155	26	47	26	14	131	10	11	16	150	12	13
9	160	163	28	51	28	15	131	12	12	17	150	13	14

単位：秒

テスト内容
1：VOLCOPY 5 実行時間（FM-11）*1
2：VOLCOPY実行時間（FM-7）*1
3：FM-11　ブート時間
4：4号表紙サービス実行時間（24Kbyte×4のマシン語ロード）（FM-11）
5：4号表紙サービス½のロード時間（24Kbyte×2のマシン語ロード）（FM-7）
6：6号トッカータとフーガ　セーブ時間（バイナリ）（FM-11）
7：　〃　〃（〃）（FM-7）
8：　〃　ロード時間（〃）（FM-11）
9：　〃　〃（〃）（FM-7）
10：　〃　セーブ時間（ASCII）（FM-11）
11：　〃　〃（〃）（FM-7）
12：　〃　ロード時間（〃）（FM-11）
13：　〃　〃（〃）（FM-7）

※　セクタの番号以外のフォーマットはSYSDSKプログラムでフォーマットされたものと同一の時のケースです。
例えば，次の番号のセクタが隣にある通常のフォーマットでもFMのSYSDSKプログラムでフォーマットしたものと処理時間がかなり異なる場合があります（NECのPCでフォーマットしたもの，FMに初めから付いているシステムディスクなど）。

*1　普通のフォーマットからのコピー時間です。

リスト1　FORMAT.B

C 1983　T.Tsuruoka

Add	+0	+1	+2	+3	+4	+5	+6	+7	Sum
4C00	34	1B	1A	50	32	7D	33	E4	:7F
4C08	86	FD	1F	8B	A6	03	30	8D	:93
4C10	00	EE	A7	07	86	08	A7	84	:55
4C18	AD	9F	FB	FA	25	2E	6F	C4	:C7
4C20	6F	41	0F	1C	8E	4F	00	8D	:45
4C28	33	26	2B	6C	41	0C	1C	8D	:E6
4C30	2B	26	23	A6	C4	81	27	27	:AD
4C38	07	6C	C4	17	00	A1	20	E0	:EF
4C40	10	8E	00	00	32	63	35	1B	:83
4C48	10	AF	02	39	E6	8D	00	B1	:1E
4C50	86	00	1F	02	20	EE	86	80	:BB
4C58	1F	02	20	E8	10	8E	50	00	:17
4C60	CC	4E	20	8D	74	86	10	A7	:78
4C68	42	8D	6B	CC	F5	03	8D	69	:F4
4C70	86	FE	A7	A0	A6	C4	A7	A0	:7C
4C78	A6	41	A7	A0	A6	80	A7	A0	:9B
4C80	86	01	A7	A0	86	F7	A7	A0	:92
4C88	CC	4E	16	8D	4C	8D	47	CC	:A9
4C90	F5	03	8D	45	86	FB	A7	A0	:92
4C98	CC	E5	00	8D	3C	86	F7	A7	:9E
4CA0	A0	CC	4E	36	8D	33	6A	42	:5C
4CA8	26	BF	CC	4E	00	8D	2A	CC	:82
4CB0	4E	00	8D	25	CC	4E	00	8D	:A7
4CB8	20	10	8E	50	00	86	F4	97	:1F
4CC0	18	A6	A0	D6	1F	2A	04	97	:18
4CC8	1B	20	F6	C5	40	27	F4	D6	:27
4CD0	18	8D	27	C4	E4	39	CC	00	:79
4CD8	0C	A7	A0	5A	26	FB	39	8D	:94
4CE0	09	86	5A	97	18	8D	13	8D	:C5
4CE8	08	39	96	18	85	01	26	FA	:95
4CF0	39	96	1F	85	40	27	FA	96	:6A
4CF8	18	39	86	14	4A	26	FD	39	:91

リスト2　リスト1のアセンブラソースリスト

```
01000                              NAM    FORMAT
01010                        *     Oh! FM DISK UTILITYfor DISK FORMAT
01020                        *     Last edition 1983.12.4
01030                              OPT    S
01040  4C00                        ORG    $4C00
01050            0018   FDCCRG EQU    $0018
01060            0019   TRKREG EQU    $0019
01070            001A   SCTREG EQU    $001A
01080            001B   FDCDRG EQU    $001B
01090            001C   HREG   EQU    $001C
01100            001D   DRVREG EQU    $001D
01110            001F   DRQIRQ EQU    $001F
01120            FBFA   BIOS   EQU    $FBFA
01130            0000   TRK    EQU    0
01140            0001   SIDE   EQU    1
01150            0002   SCT    EQU    2
01160            5000   SDTTOP EQU    $5000
01170            4F00   SCDATA EQU    $4F00
01180            0041   WAITD  EQU    65
01190            4C00   ENTRY  EQU    *
01200  4C00 34   1B            PSHS   CC,DP,A,X           レジスタ退避
01210  4C02 1A   50            ORCC   #$50     ;MASK IRQ&FIRQ
01220  4C04 32   7D            LEAS   -3,S     ;LINK work 3 bytes
01230  4C06 33   E4            LEAU   ,S                  ワークエリアのポインタにUを使用
01240  4C08 86   FD            LDA    #$FD
01250  4C0A 1F   8B            TFR    A,DP                ダイレクトページをI/Oエリアに
01260  4C0C A6   03            LDA    3,X                 ユーザー関数で渡されたドライブNo.ロード
01270  4C0E 30   8D 00EE       LEAX   RCB,PCR             XレジスタにRCBをセット
01280  4C12 A7   07            STA    7,X      ;set drv No.        ドライブNo.をRCB+7にセット
01290  4C14 86   08            LDA    #$08     ;RESTORE command  } リストアのコードをRCBの先頭にセット
01300  4C16 A7   84            STA    ,X                          }
01310  4C18 AD   9F FBFA       JSR    [BIOS]              BIOSコール(ドライブセレクト,トラック0移動)
01320  4C1C 25   2E   4C4C     BCS    ERROR               エラーチェック
01330  4C1E 6F   C4            CLR    TRK,U               トラックNo.を0に
01340            4C20   CYLLP  EQU    *
01350  4C20 6F   41            CLR    SIDE,U              サイド番号を0に
01360  4C22 0F   1C            CLR    HREG     ;SET HEAD TO SIDE   ヘッド番号0に
01370  4C24 8E   4F00          LDX    #SCDATA             セクタナンバのデータのトップアドレスをXに
01380  4C27 8D   33   4C5C     BSR    TRKWR               1トラック フォーマット
01390  4C29 26   2B   4C56     BNE    ERROR2              エラーならエラー処理
01400  4C2B 6C   41            INC    SIDE,U              サイド番号を+1
01410  4C2D 0C   1C            INC    HREG                ヘッド番号を+1
01420  4C2F 8D   2B   4C5C     BSR    TRKWR               1トラック フォーマット
01430  4C31 26   23   4C56     BNE    ERROR2              エラーならエラー処理
01440  4C33 A6   C4            LDA    TRK,U             } 39トラック終了か
01450  4C35 81   27            CMPA   #39               }
01460  4C37 27   07   4C40     BEQ    NORMAL
01470  4C39 6C   C4            INC    TRK,U               トラックNo.を1つ増やす
01480  4C3B 17   00A1 4CDF     LBSR   STEPIN              1つ内側のトラックへ
01490  4C3E 20   E0   4C20     BRA    CYLLP               1トラック(表と裏)フォーマット繰り返す
01500            4C40   NORMAL EQU    *
01510  4C40 108E 0000          LDY    #$0000              ユーザー関数として0を返す(正常)
01520  4C44 32   63     ENDFMT LEAS   3,S      ;UNLINK    スタックを元に戻す
01530  4C46 35   1B            PULS   CC,DP,A,X
01540  4C48 10AF 02            STY    2,X
01550  4C4B 39                 RTS
01560            4C4C   ERROR  EQU    *
01570  4C4C E6   8D 00B1       LDB    RCBSTA,PCR
```

```
4C50 86   00                  LDA    #$00
4C52 1F   02                  TFR    D,Y
4C54 20   EE   4C44           BRA    ENDFMT
          4C56      ERROR2 EQU    *
4C56 86   80                  LDA    #$80
4C58 1F   02                  TFR    D,Y
4C5A 20   E8   4C44           BRA    ENDFMT
                    *         write one track
          4C5C      TRKWR  EQU    *
4C5C 108E 5000                LDY    #SDTTOP
4C60 CC   4E20                LDD    #$4E20   ;GAP 1
4C63 8D   74   4CD9           BSR    SETDT
4C65 86   10                  LDA    #16
4C67 A7   42                  STA    SCT,U
4C69 8D   6B   4CD6 SCTLP  BSR    SYNC
4C6B CC   F503                LDD    #$F503   ;AM
4C6E 8D   69   4CD9           BSR    SETDT
4C70 86   FE                  LDA    #$FE     ;AM
4C72 A7   A0                  STA    ,Y+
4C74 A6   C4                  LDA    TRK,U    ;ID C (TRACK No.)
4C76 A7   A0                  STA    ,Y+
4C78 A6   41                  LDA    SIDE,U   ;ID H (SIDE No.)
4C7A A7   A0                  STA    ,Y+
4C7C A6   80                  LDA    ,X+      ;ID R (SECTOR No.)   ……セクタ番号をセクタナンバデータから取ってくる
4C7E A7   A0                  STA    ,Y+
4C80 86   01                  LDA    #1       ;ID N (DATA LENGTH)  1トラック フォーマットデータ作成
4C82 A7   A0                  STA    ,Y+
4C84 86   F7                  LDA    #$F7     ;ID CRC
4C86 A7   A0                  STA    ,Y+
4C88 CC   4E16                LDD    #$4E16   ;GAP2
4C8B 8D   4C   4CD9           BSR    SETDT
4C8D 8D   47   4CD6           BSR    SYNC
4C8F CC   F503                LDD    #$F503   ;DM
4C92 8D   45   4CD9           BSR    SETDT
4C94 86   FB                  LDA    #$FB     ;DM
4C96 A7   A0                  STA    ,Y+
4C98 CC   E500                LDD    #$E500   ;1 SECTOR DATA
4C9B 8D   3C   4CD9           BSR    SETDT
4C9D 86   F7                  LDA    #$F7     ;CRC
4C9F A7   A0                  STA    ,Y+
4CA1 CC   4E36                LDD    #$4E36   ;GAP3
4CA4 8D   33   4CD9           BSR    SETDT
4CA6 6A   42                  DEC    SCT,U
4CA8 26   BF   4C69           BNE    SCTLP
4CAA CC   4E00                LDD    #$4E00   ;GAP4
4CAD 8D   2A   4CD9           BSR    SETDT
4CAF CC   4E00                LDD    #$4E00   ;GAP4
4CB2 8D   25   4CD9           BSR    SETDT
4CB4 CC   4E00                LDD    #$4E00   ;GAP4
4CB7 8D   20   4CD9           BSR    SETDT
                    *         write command
4CB9 108E 5000                LDY    #SDTTOP
4CBD 86   F4                  LDA    #$F4     ;write track with E(HLDdelay)
4CBF 97   18                  STA    FDCCRG
                                              FDCに1トラック フォーマット命令を送る
4CC1 A6   A0        DSET   LDA    ,Y+
4CC3 D6   1F        DWAIT  LDB    DRQIRQ
4CC5 2A   04   4CCB           BPL    NOTDRQ
4CC7 97   1B                  STA    FDCDRG
4CC9 20   F6   4CC1           BRA    DSET
4CCB C5   40        NOTDRQ BITB   #$40     ;TEST IRQ           FDCにフォーマット用データを送る
4CCD 27   F4   4CC3           BEQ    DWAIT
4CCF D6   18                  LDB    FDCCRG   ;READ STATUS
4CD1 8D   27   4CFA           BSR    WAIT5    ;wait 50Ms
4CD3 C4   E4                  ANDB   #$E4     ;CHECK NR,WP,WF,LD
4CD5 39                       RTS
4CD6 CC   000C      SYNC   LDD    #$000C
4CD9 A7   A0        SETDT  STA    ,Y+
4CDB 5A                       DECB
4CDC 26   FB   4CD9           BNE    SETDT
4CDE 39                       RTS
                    *         step in sub routine
          4CDF      STEPIN EQU    *
4CDF 8D   09   4CEA           BSR    WREADY
4CE1 86   5A                  LDA    #$5A     ;STEPIN,sr20ms,h,u
4CE3 97   18                  STA    FDCCRG                      1トラック内側に移動するコマンドを
4CE5 8D   13   4CFA           BSR    WAIT5                       FDCに送る
4CE7 8D   08   4CF1           BSR    WIRQ
4CE9 39                       RTS
                    *         tiny sub routines
4CEA 96   18        WREADY LDA    FDCCRG
4CEC 85   01                  BITA   #$01                        FDCのレディ待ち
4CEE 26   FA   4CEA           BNE    WREADY
4CF0 39                       RTS
4CF1 96   1F        WIRQ   LDA    DRQIRQ
4CF3 85   40                  BITA   #$40
4CF5 27   FA   4CF1           BEQ    WIRQ                        FDCの命令実行終了待ち
```

```
02440  4CF7 96   18               LDA    FDCCRG
02450  4CF9 39                    RTS
02460  4CFA 86   14        WAIT5  LDA    #20       ;wait 2.5*20=50Ms
02470  4CFC 4A             WTLP2  DECA
02480  4CFD 26   FD   4CFC        BNE    WTLP2
02490  4CFF 39                    RTS
02500  4D00      0008      RCB    RMB    8
02510            4D01      RCBSTA EQU    RCB+1
02520            4C00             END    ENTRY
TOTAL ERRORS 00000--00000
TOTAL WARNINGS 00000--00000

PROGRAM BEGIN ADDR=4C00
PROGRAM END   ADDR=4D07
PROGRAM ENTRY ADDR=4C00


BIOS   FBFA  CYLLP  4C20  DRQIRQ 001F  DRVREG 001D  DSET   4CC1  DWAIT  4CC3
ENDFMT 4C44  ENTRY  4C00  ERROR  4C4C  ERROR2 4C56  FDCCRG 0018  FDCDRG 001B
HREG   001C  NORMAL 4C40  NOTDRQ 4CCB  RCB    4D00  RCBSTA 4D01  SCDATA 4F00
SCT    0002  SCTLP  4C69  SCTREG 001A  SDTTOP 5000  SETDT  4CD9  SIDE   0001
STEPIN 4CDF  SYNC   4CD6  TRK    0000  TRKREG 0019  TRKWR  4C5C  WAIT5  4CFA
WAITD  0041  WIRQ   4CF1  WREADY 4CEA  WTLP2  4CFC
```

約50μs時間待ち

RCB

リスト3

```
100 CLEAR 300,&H4BFF:DEFINT A-Z
110 LOADM"FORMAT.B"
120 DEFUSR=&H4C00
130 INPUT "SKEW FACTOR(0) OR SECTOR SKIP(1)";MD
140 IF MD>2 OR MD<0 THEN 130
150 INPUT "DRIVE No.=";DN
160 ON MD+1 GOSUB 220,300
170 INPUT "OK?";A$
180 IF A$<>"y" AND A$<>"Y" THEN 150
190 A=USR(DN)
200 IF A<>0 THEN PRINT"DISK I/O ERROR"
210 END
215 'skew factor
220 INPUT "SKEW FACTOR=";SF
230 PS=1
240 FOR I=1 TO 16
250    POKE &H4EFF+I,PS
260    POKE &H4F0F+I,PS
270    PS=(PS+SF-1) MOD 16+1
280 NEXT
290 RETURN
295 'sector step
300 INPUT "SECTOR SKIP=";SS
310 PS=SS
320 FOR I=1 TO 16
330    POKE &H4EFF+PS,I
340    POKE &H4F0F+PS,I
350    PS=(PS+SS) MOD 17
360 NEXT
370 RETURN
```

同じトラック番号のサイド0と1は，連続して読む機会が多いのです。そのため，このプログラムは，サイド0と1でセクタの順番を違えることができるようにしてあります。通常の使用では，&H4F00番地から&H4F0F番地までの16バイト（サイド0のセクタ番号を並べる）と，&H4F10～&H4F1F番地の16バイト（サイド1用）を同じに設定して使います。このエリアにセクタ番号1～16を好きな順に入れるわけです。

リスト3は，スキューのためのセクタ並びデータを作成し，リスト1のプログラムを呼び出すプログラムです。標準的なスキューをかけるものと，表1のテストに使用した偶数個おきに連続セクタになるものの2とおりが選べます。前者は図3のようにフォーマットされます。後者は最後のセクタと最初のセクタとの間が他よりも広いことに着目し，ここにセクタ0があると仮想して，順番を決めます。0～16の17セクタなら奇数ですから，偶数個おきも可能になり，仮想的に考えたセクタ0も実在せず，ちゃんと16個セクタができるわけです。

なお，当然フォーマットしたディスクは，DSKINIコマンドを実行するか，SYSDSKプログラムを実行（このときFORMATは不要）か，VOLCOPYで他のディスクをコピーするかしないと，F-BASICでは使えません。

特集 ディスク活用術

ディスクユーティリティ集

for FM-7 ステップレート変更プログラム

——薄型ドライブのセーブをより静かに速く——

Q-Laboratory

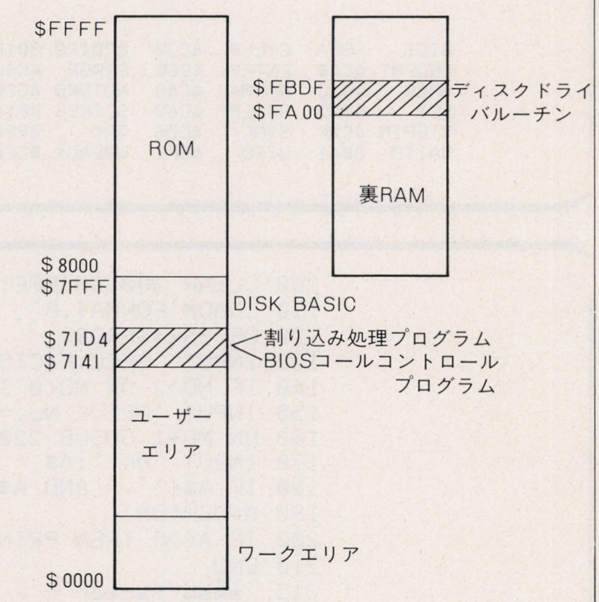
フロッピディスクドライブは，回転するディスケットからの信号の出し入れを，ディスケットの直径方向に移動するヘッドを通じて行います。このヘッドを，指定したトラックに移動させるために，フロッピディスクコントローラ(FDC)はドライブに対してSTEPと名づけられているパルス信号を送ります。ドライブは，STEP信号を１つ受け取るとヘッドを１トラック分内側または外側へ移動させるわけです（移動する方向はDIRCという信号によって決定されます)。ヘッドをN個内側または外側へ移動させる場合には，FDCはドライブにたいして等間隔のN個のSTEP信号を与えますが，１つのSTEP信号と次のSTEP信号との間隔をステップレート(STEP-RATE)と呼びます。

このステップレートは，FDCに与えるコマンドによって30，20，12，6 ms（クロック１MHzの場合）の４段階に変更が可能であり，今回のプログラムは，6 msのステップレート（ディフォルト値は20 ms）でディスクドライブを駆動するためのものです。

ディスケットへの書き込みの場合，F-BASIC (V 3.0)ではヘッドの移動に多くの時間を使っています。このプログラムを実行すると，SAVEに必要とする時間が約半分に減少しますので確かめてみてください。

なお，ステップレートの規格が6msを超えるディスクドライブでは，このプログラムのまま実行させるとシークエラーが

図１　イニシャライズ後のメモリマップ



リスト１　STARTUPB

©1983 Q-Laboratory

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
3000  1A 50 8E FE 00 CE 40 00 :04
3008  10 8E 02 00 17 00 A5 8E :EA
3010  40 02 86 16 CE 00 5C 17 :1F
3018  00 A3 CE 01 56 17 00 9D :7C
3020  CE 01 53 17 00 97 86 08 :5E
3028  B7 40 82 86 1C B7 40 9E :B0
3030  86 17 CE 00 B2 8E 40 7B :66
3038  17 00 82 CE 00 9A 8E 40 :CF
3040  B6 8D 7A CE FF 56 8E 41 :AF
3048  0D 8D 72 CE FF 24 8E 41 :CC
3050  5E 8D 6A 8E 40 00 CE FA :EB
3058  00 10 8E 01 E0 7F FD 0F :0A
3060  8D 52 CE FF E0 10 8E 00 :2A
3068  20 8D 49 10 8E 00 93 34 :5B
3070  20 FC 05 9D 1F 01 A3 E1 :62
3078  1F 03 83 00 02 FD 05 9D :46
3080  DD 45 83 01 2C DD 3F 1F :0D
3088  04 34 10 30 8D 00 62 8D :F4
3090  23 7D FD 0F CC FF 6E E3 :C8
3098  E4 DD DF FD FB FA CC FF :5D
30A0  BE E3 E1 FD 01 E1 FD 01 :5F
30A8  DE BD 8F 39 BD 9B 50 1C :27
30B0  AF 7E 86 3E A6 80 A7 C0 :7E
30B8  31 3F 26 F8 39 A7 80 EF :DD
30C0  81 39 43 68 61 6E 67 65 :00
30C8  20 53 74 65 70 2D 72 61 :BC
30D0  74 65 20 43 6F 70 79 72 :06
30D8  69 67 68 74 28 63 29 20 :80
30E0  31 39 38 33 20 51 2D 4C :BF
30E8  61 62 6F 72 61 74 6F 72 :5A
30F0  79 00 34 7F 6F 8C FA A6 :C7
30F8  84 81 08 25 3D 81 0A 22 :1C
3100  39 1A 50 63 8C EB 7F FD :F9
3108  0F 35 01 34 01 86 FD 1F :1C
3110  8B 10 8E FA 08 A6 84 81 :D6
3118  08 26 02 31 3A 34 10 AD :8C
3120  A4 35 10 35 01 A7 01 26 :ED
3128  03 1C FE 8E 1A 01 35 7E :79
3130  34 01 6F 8C BC 7D FD 0F :75
3138  35 81 7D FD 0F 35 7F 7E :71
3140  F1 7D 34 57 7D FD 0F C6 :48
3148  08 8E C9 53 A6 E4 C5 80 :81
3150  27 05 C6 11 8E D2 FC AF :0E
3158  65 6D 8C 95 27 15 32 7E :DF
3160  1F 41 34 04 A6 02 A7 80 :67
3168  5A 26 F9 35 04 30 8D 00 :6F
3170  04 AF E5 35 97 34 03 1A :B5
3178  50 86 FF A7 8D FF 72 7F :F9
3180  FD 0F 35 83 00 00 00 00 :C4
```

発生する可能性があります。ディスクドライブのカタログを調べて，ステップレート（トラック間シーク速度）規格が 12ms のドライブについては，＄3027番地を＄８→＄9に，＄302C番地の値を＄１C→＄１D に変更して使用し，ステップレートの規格が15，20msなど12msを超えるドライブについては，このプログラムは使用しないでください。

プログラムは大きく分けて３つのルーチンよりなっています。１つは，フロッピディスクドライバルーチンの書き換えを行ったうえで裏RAMに転送したり，フロッピディスク関係の BIOS ルーチンや割り込み処理ルーチンをユーザー領域の最後部にセットし，それに応じたポインタ類の再設定を行う，イニシャライズルーチンです。ドライバルーチンの書き換えを行っているのは，①ヘッド駆動関係コマンドの部分，②ドライバルーチンの格納番地の変更が必要なため，絶対ジャンプ命令から相対ジャンプ命令に変更が必要な部分です。

２つ目のルーチンは，フロッピディスク関係のBIOSコールの場合，裏RAMに転送したドライバルーチンをコールするためのプログラムです。

３つ目のルーチンは，FIRQ，IRQがあった場合の処理ルーチンです。裏RAM のプログラムを実行中にFIRQやIRQの割り込みがあった場合には，RAM モードからROMモードに切り換えて，割り込み処理ルーチンに制御を移すとともに，割り込み処理から戻った場合には，再度裏RAMのプログラムに制御を戻してやる必要がありますが，この処理を行うのがこのルーチンの目的です。

このプログラム実行後のアドレスマップを図１に示しました。なお，F-BASICプログラムはもちろん，機械語プログラムであっても＄7141番地以前に格納されるものであれば，従来と変わりなく動作します。

リスト１に機械語プログラムを，**リスト２**にBASICのローダを示します。このプログラムを STARTUP ファイルとして使用すると，ディスクBASIC を立ち上げるとともに，必要な初期化が行われますので，便利に使えると思います。

リスト２　STARTUP

```
10 LOADM"STARTUPB",,R
```

リスト３　リスト１の割り込み処理部ソースリスト

```
                        **********************
                        *
                        * FIRQ,IRQ ﾉ ｼｮﾘ
                        *

3142 34    57           CIRQ  PSHS  CC,A,B,X,U
3144 7D    FD0F               TST   MODE       ROMﾓｰﾄﾞ ﾆ ｽﾙ
3147 C6    08                 LDB   #8
3149 8E    C953               LDX   #FIRQ
314C A6    E4                 LDA   ,S
314E C5    80                 BITB  #$80       FIRQ ?
3150 27    05                 BEQ   IRQ1       YES
3152 C6    11                 LDB   #17        NO IRQ
3154 8E    D2FC               LDX   #IRQ
3157 AF    65           IRQ1  STX   5,S
3159 6D    8C 95              TST   FLAG,PCR   RAM ﾓｰﾄﾞ ?
315C 27    15                 BEQ   JMPI       NO
315E 32    7E                 LEAS  -2,S       YES
3160 1F    41                 TFR   S,X
3162 34    04                 PSHS  B
3164 A6    02           LOOP  LDA   2,X
3166 A7    80                 STA   ,X+
3168 5A                       DECB
3169 26    F9                 BNE   LOOP
316B 35    04                 PULS  B
316D 30    8D 0004            LEAX  RTIRQ,PCR ｲﾝﾀﾗﾌﾟﾄ ｶﾗﾉ ﾓﾄﾞﾘ ﾊﾞﾝﾁ
3171 AF    E5                 STX   B,S
3173 35    97           JMPI  PULS  CC,A,B,X,PC IRQ ﾙｰﾁﾝ ﾍ ｼﾞｬﾝﾌﾟ

                        ***********************************
                        *
                        *  ﾜﾘｺﾐﾏｴ ﾉ ﾌﾟﾛｸﾞﾗﾑ ﾆ ﾓﾄﾞﾙ
                        *

3175 34    03           RTIRQ PSHS  CC,A
3177 1A    50                 ORCC  #$50
3179 86    FF                 LDA   #$FF
317B A7    8D FF72            STA   FLAG,PCR
317F 7F    FD0F               CLR   MODE       RAMﾓｰﾄﾞ ﾆ ｽﾙ
3182 35    83                 PULS  CC,A,PC

                   3184 TAIL  EQU   *
                              END   START
```

特集 ディスク活用術

ディスクユーティリティ集

for FM-7,8,11 カタログ表示プログラム&ディスケット再利用プログラム

山岸英人

ディスクを使用する際に，ちょっと役に立つディスクユーティリティを，2つ紹介しましょう。

＊　＊　＊

1. カタログ表示プログラム

ディスケット上に現在，どのようなプログラムやファイルがあるかを調べるためにFILES という命令がありますが，これは割と不親切なもので，もっといろいろな機能が欲しくなります。そこで今回は，このFILES 命令の機能を拡張させるプログラムを作りました。

使い方は，RUN させた後，S（スクリーン表示）かL（プリンタ出力）を入力し，ディスクのドライブ番号の入力をするだけで，ファイル名，ファイルの種類と属性，占有しているクラスタ数ではなく実際に使用しているクラスタ番号を表示します。その際，最初のクラスタ番号のみ，トラック番号とセクタ番号を16進で表示します。また，マシン語なら開始番地，終了番地，入口番地を表示します。ESCキーで止めながら見てください。

カタログ表示プログラム

©1983 H.YAMAGISHI

```
10 ´
20 ´<<< CATAROG >>>
30 ´
40 CLEAR4096,&H4FFF
50 FIELD#0,128 AS A$(0),128 AS A$(1)
60 DEF FNS$(N)=MID$(STR$(N),2,LEN(STR$(N))-1)
70 E$=STRING$(8,&HFF):TP$(0)=" BASIC ":TP$(1)=" DATA  ":TP$(2)=" M-CODE"
80 AB$(1)=" A":AB$(0)=" B":RS$(1)=" R":RS$(0)=" S"
90 INPUT"SCRN: or LPT0: ? (S/L) ",D$:IF D$="L" OR D$="1" THEN OPEN"O",#1,"LPT0:"
 ELSE OPEN"O",#1,"SCRN:"
100 PRINT
110 PRINT#1:PRINT#1,"<<< CATALOG >>>":PRINT#1
120 INPUT"DRIVE:",DRV:PRINT
130 D$=DSKI$(DRV,1,1)
140 FOR I=0 TO 255:N=INT(I/128):POKE &H5000+I,ASC(MID$(A$(N),(I+1)-N*128,1)):NEX
T I
150 SEC=4
160 D$=DSKI$(DRV,1,SEC)
170 N=0:I=1
180 NA$=MID$(A$(N),I,8)
190 IF NA$=E$ THEN END
200 IF ASC(MID$(NA$,1,1))=0 THEN 370
210 PRINT#1,NA$;
220 TP=ASC(MID$(A$(N),I+11,1))
230 PRINT#1,TP$(TP);
240 AB=ASC(MID$(A$(N),I+12,1))
250 IF AB=255 THEN AB=1
260 PRINT#1,AB$(AB);
270 RS=ASC(MID$(A$(N),I+13,1))
280 IF RS=255 THEN RS=1
290 PRINT#1,RS$(RS);
300 CL=ASC(MID$(A$(N),I+14,1))
310 PCL=CL:ECL=CL:PRINT#1," cluster";FNS$(CL);"(T";RIGHT$("00"+HEX$(CL/4+2),2);"
 S";RIGHT$("00"+HEX$((CL MOD 4)*8+1),2);")";
320 ECL2=CL:CL=PEEK(&H5005+CL)
330 IF CL<&HC0 THEN PRINT#1,".";FNS$(CL);:ECL=CL:GOTO320 ELSE ESEC=CL-&HBF
340 PRINT#1
350 IF TP=2 THEN GOSUB400
360 PRINT#1
370 I=I+32:IF I=129 THEN I=1:N=N+1
380 IF N=2 THEN SEC=SEC+1:GOTO160
390 GOTO180
400 D$=DSKI$(DRV,2+INT(PCL/4),(PCL MOD4)*8+1)
410 SA=ASC(MID$(A$(0),4,1))*256+ASC(MID$(A$(0),5,1))
420 EA=ASC(MID$(A$(0),2,1))*256+ASC(MID$(A$(0),3,1))
430 PRINT#1,"          addr";" &H"+RIGHT$("0000"+HEX$(SA),4);
440 PRINT#1,",&H"+RIGHT$("0000"+HEX$(SA+EA-1),4);
450 AD=(EA MOD256)+8:IF AD>=256 THEN AD=AD-256
460 IF AD=255 THEN 520
470 D$=DSKI$(DRV,2+INT(ECL/4),(ECL MOD4)*8+ESEC)
```

```
480 AN=INT(AD/128):AI=AD MOD128
490 XA=ASC(MID$(A$(AN),AI+1,1))*256+ASC(MID$(A$(AN),AI+2,1))
500 PRINT#1,",&H"+RIGHT$("0000"+HEX$(XA),4)
510 D$=DSKI$(DRV,1,SEC):RETURN
520 ESEC=ESEC-1:IF ESEC=0 THEN D$=DSKI$(DRV,2+INT(ECL2/4),(ECL2 MOD4)*8+8):HD=AS
C(MID$(A$(1),128,1)):D$=DSKI$(DRV,2+INT(ECL/4),(ECL MOD4)*8+1):LD=ASC(MID$(A$(0)
,1,1)):XA=HD*256+LD:GOTO500
530 D$=DSKI$(DRV,2+INT(ECL/4),(ECL MOD4)*8+ESEC):HD=ASC(MID$(A$(INT(AD/128)),(AD
 MOD128)+1,1)):D$=DSKI$(DRV,2+INT(ECL/4),(ECL MOD4)*8+ESEC+1):LD=ASC(MID$(A$(INT
(AD/128)),(AD MOD128)+1,1)):XA=HD*256+LD:GOTO500
```

2.ディスケット再利用プログラム

このプログラムは，壊れたディスケット（傷などがついてしまいDevice I/O Errorが出るもの，すでに寿命が尽きてBad File Structureなどが出始めたものなど）の使用不能なったクラスタを使用禁止にして再利用しようというものです。

使い方は，プログラムをロード後，壊れてしまったディスケットをドライブ0に入れてRUNすると，ディスケットのチェックを始めます。ディスケット上のファイルを壊さずにチェックしたい場合は1，壊してもいい場合は2を入力してください。ただし，READ ONLYのチェックではひっかからない傷などがありますので，注意してください。

チェックが終わるとディレクトリを書き換えるかどうか聞いてきますので，YかNを入力します。書き換える場合は，あらかじめDSKINIを実行したディスケットにしてください。

ディスケット再利用プログラム（FM—11の方は色文字のように変更してください）

©1983 H. YAMAGISHI

```
10 '
20 '<<< DISKET CHECK & RECYICLE >>>
30 '
40 COLOR7,0:WIDTH80,25:CLEAR4096,&H4FFF:DIM EC(151),ET(151),ES(151):CO$=CHR$(17)
:UP$=CHR$(30):BK$=CHR$(29):TABLE=&H5000
50 FIELD#0,128 AS A$(0),128 AS A$(1)
60 ON ERROR GOTO370
70 PRINT:PRINTCO$+"6K<<< DISKET CHECK >>>":PRINT
80 '
90 '<<< LOAD DIR >>>
100 '
110 PRINTCO$+"4LOADING DIR...":PRINT:PRINT:D$=DSKI$(0,1,1)
120 FOR N=0 TO 1:FOR I=0 TO 127
130 V=ASC(MID$(A$(N),I+1,1)):CL=N*128+I:POKE TABLE+CL,V:CL=CL-5:IF CL>151 THEN 1
60
135 IF CL<0 THEN 160
140 PRINTUP$+CO$+"1CLUSTER";CL,RIGHT$("00"+HEX$(V),2);" ";:IF V=&HFD OR V=&HFF T
HEN PRINTCO$+"3ENPTY ":GOTO160
150 IF V=&HFE THEN PRINTCO$+"4SYSTEM":GOTO160 ELSE PRINT"USED "
160 NEXT I,N
170 '
180 '<<< CHECK DISKET >>>
190 '
200 PRINT:PRINTCO$+"4CHECK DISKET...":PRINT:PRINT"1:READ ONLY  2:READ WRITE  ";
210 INPUT"(1/2):",CMD$:E=0:PRINT:PRINT
220 FOR CL=0 TO 151:TRK=2+INT(CL/4):SECTOP=(CL MOD 4)*8
230 PRINTUP$+CO$+"1CLUSTER";CL;"TRACK:";RIGHT$("00"+HEX$(TRK),2);" SECTOR:  ";
240 FOR SEC=1 TO 8:PRINTBK$+BK$+RIGHT$("00"+HEX$(SECTOP+SEC),2);:D$=DSKI$(0,TRK,
SECTOP+SEC):IF CMD$="2" THEN DSKO$ 0,TRK,SECTOP+SEC
250 NEXT SEC:PRINT
260 NEXT CL
270 '
280 '<<< REMAKE >>>LIST
290 '
300 IF E=0 THEN END
310 PRINT:PRINTE;CO$+"2CLUSTERS CRASHED!!":FOR I=1 TO E:PRINT"CLUSTER";EC(E);"TR
ACK:";RIGHT$("00"+HEX$(ET(E)),2);" SECTOR:";RIGHT$("00"+HEX$(ES(E)),2):NEXT I:PR
INT
320 INPUT"REMAKE DIR? (Y/N):",CMD$:IF CMD$<>"Y" THEN END
330 FOR N=0 TO 1:FOR I=0 TO 127:CL=N*2+I
340 B$(N)=B$(N)+CHR$(PEEK(TABLE+CL))
350 NEXT I,N
360 LSET A$(0)=B$(0):LSET A$(1)=B$(1):DSKO$ 0,1,1:FILES:END
370 '
380 '<<< CRASHED CLUSTER >>>
390 '
400 PRINTCO$+"2 CRASHED!!";:BEEP:FOR I=1 TO 500:NEXT I:PRINTSTRING$(10,BK$);STRI
NG$(10," ")
410 POKE TABLE+5+CL,&HC7:E=E+1:EC(E)=CL:ET(E)=TRK:ES(E)=SECTOP+SEC
420 RESUME 260
```

for FM-7

F-BASICに裏RAMコマンドを拡張
オールRAMモード活用法

是方　研二（神戸大学電子考古学研究所）

プロローグ

FM-7にはF-BASICでは使われていない裏RAMというメモリが存在します。現在のF-BASICではこのメモリをアクセスすることはできませんが，FM-7を徹底的に使いこなそうとするユーザーとしては，これを見逃すわけにはいきません。

そこで，BASICレベルで，この裏RAMを自由に操作できるようF-BASICを拡張し，FM-7の潜在能力をばっちり引き出そうというわけです。

オールRAMモードとは

買ったばかり……という初心者のために裏RAMとは何かということを説明します。そもそもFM-7のメインブロックのメモリ構成は図1の2つのモードがあって，ディップスイッチで①②のモードが設定されます。通常F-BASICを使っているときは①のモードになっていて，②のモードを我々はオールRAMモードと呼んでいます。このモードにすると，①でROMのあった場所にRAMが置き換えられ，このRAMを裏RAMと呼びます。FM-7では この①②モード切り換えをディップスイッチの設定のほかに，マシン語レベルでソフト的に行うことが可能です（ここでFM-8のユーザーは，ガツーンとshock！）。マシン語派の人には1つ注意してほしいのですが，「ユーザーズマニュアルシステム仕様」の1-25にある裏RAMの説明はかなり曖昧で，①のモードを32K RAM＋32K ROMと書いてありますが，図1を見てのとおり＄8000以降にもRAMはあります。それに「切り換えは，＄0000～＄7FFF番地内のプログラムで行わなくてはなりません。＄8000番地以降のプログラムで切り換えますと暴走します」とありますが，これも＄FC00～＄FC7F内で切り換えしても暴走しません。要するにユーザーとしては＄8000～＄FBFFの内容だけが入れ換わることを考えてプログラミングすればよいのです。

さて図2を見てください。この機能を利用すると，普段はモード①にしておいて必要なときにモード②にする。そしてCからデータをAに移し，また①モードに切り換える，という抜け目ない切り換え操作で，B，C両方の領域を利用できるのです。裏RAMはなんと31Kバイト（メチャ広い）もあるので1つのCPUで64K＋31K＝95Kバイトのメモリを操作できることになります。恐るべし!! FM-7が95Kバイトマシンだったとは……！　こうやって裏RAMを使うことができれば，ビジネスソフトやワープロ，その他大量数値処理パッケージなど大量データを扱うプログラムのデータ量が飛躍的に増大します。ところがF-BASICは当然B領域のROMに書かれているので，BASICでモードに切り換えることができるとすると，自分自身がなくなってしまうという気味悪いことになります。そんなわけでF-BASICで裏RAMをアクセスするのは不可能なのです。

マシン語を知らない人はどうするか

今回，BASICを中心にプログラミンする人も容易にオールRAMモードでアクセスできるように，F-BASICに4つのコマンドと1つの関数を増設しました。まずは次のマニュアルを見てください。

図1

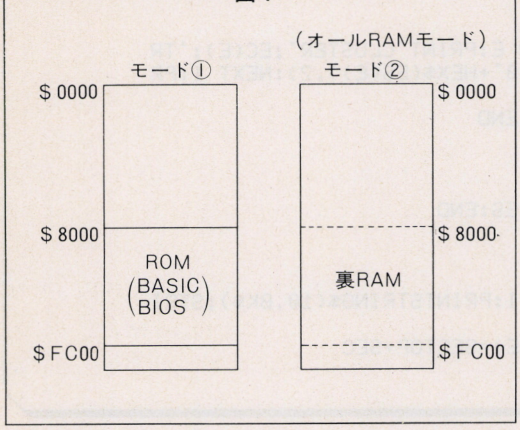
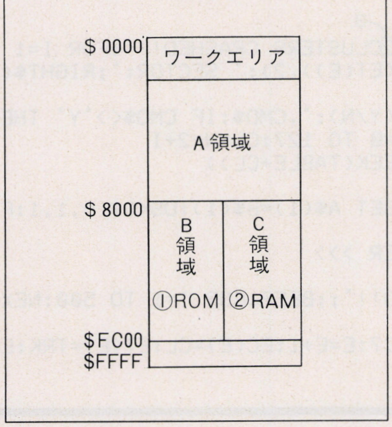


図2



《マニュアル》

◎**UPEEK**（式）

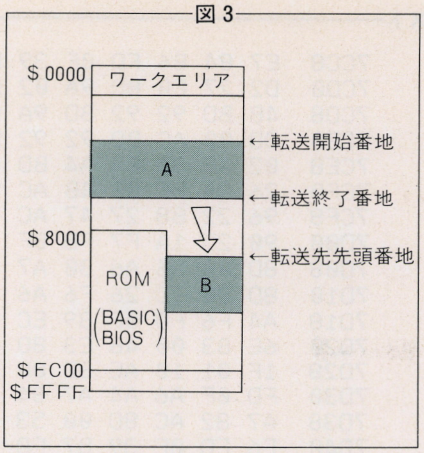
PEEK文と同じですが，オールRAMモードでメモリの値を読み込むため，＄8000～＄FBFFでは裏RAMをPEEKすることになります。

◎**UPOKE**　書き込み番地，データ

これもPOKE文と同じで，＄8000～＄FBFFでは裏RAMに書きこむことになります。

◎**TRANS**　転送開始番地，転送終了番地，転送先先頭番地

図3



ある領域のメモリの値をごっそりと，ある領域へ転送する命令です。

図3は，Aにあるデータ，あるいはプログラムをB領域に転送する場合を示しています。オールRAMモードで転送しますから，裏RAMとのデータの出し入れが可能です。また，移動場所は元の場所と重なってもOK!! 1バイトの転送もできます。ただし(転送開始番地≦転送終了番地)でないと，Illegal Function Callエラーとなります。

◎**UEXEC** 開始番地

プログラムをオールRAMモードで走らせるコマンドです。F-BASICもBIOSもありませんから，$0000～$7FFF内のプログラムだからといって，ROMサブルーチンをコールするプログラムやBIOSを利用するプログラムは，本コマンドでは走りません。裏RAMにあるプログラムを走らせるときに便利です。なお開始番地は省略できません。あしからず。

◎**UCLR**

裏RAMにはスイッチON直後$FF,そのほかいろいろな値が入ってます。$0～$7FFFの領域ではコールドスタートする際に$00でクリアされるのに，裏RAMはクリアされないようです。このままでは裏RAMをのぞいたときに，どこからどこまでが自分の操作したデータかがわかりにくいので，この命令で裏RAMを瞬時にクリアします。

たった5つの追加ですが，これだけあればオールRAMモードでのあらゆる操作ができるのです。とにかくインプットしてみてください。

```
CLEAR, &H7C00:EXEC &H7C00
```

で拡張されます。本プログラムはリロケータブルですから，ディスクシステムで使用される方は$6000番地台に移動させてください。ディスクBASIC上でも全く同様に使えます。これらの命令は実行するときもプログラミングするときも，本プログラムを実行してからにしてください。またプロテクトがかけてあっても関係なく働きます。命令の頭についているUはURA(裏)のUと思ってね!さてプログラムの説明はこれぐらいにして，“何ができるか”について話を進めていきたいと思います。

オールRAMモード活用法

その1) まずはオールRAMモニタを作ろう!

世間にはBASICで書かれたチェックサム，メモリチェンジなどの機能をもつモニタが各種発表されています。Oh! FMでも第3号にて“Oh!FM用チェックサムプログラム”が発表されているので，本プログラムを実行してから，チェックサムプログラムリストのPEEK文，POKE文の頭にUを付けてUPEEK,UPOKEとしてください。ご苦労様。これでオールRAMモニタのできあがり!さっそく裏RAMのダンプリストでもだしてみてください。きれいでしょう。またTRANS命令で何かを裏RAMに転送し，ちゃあーんと転送されているか,あるいはUCLRではクリアされているかどうか，オールRAMモニタで確かめてください。これがあると，自由に覗けるので，うーんとオールRAMモードが身近なものになるみたい。BASICで扱う不思議な感触を味わってください。

その2) プログラムを裏RAMに入れて走らせる

これが本稿最大の醍醐味なのだ。これをやるとフリーエリアが広々と使えるので，オールRAMモードで走るプログラムはなんでも裏RAMで走らせようというわけなのだが，本当の利点は次の性質を利用します。

ア) 本プログラムはリセットしても短いのですぐにロードしなおせる。

イ) ROMモードのときに暴走しても，リセットしても，電源を切らない限り，裏RAMのデータは消えない。つまり裏RAMは暴走に対する退避場所としての価値があるのです。

この利点をフルに生かせるのがアセンブラでしょう。オブジェクトをためしに走らせたりするとき，暴走はつきもので，そのたびにリセット，アセンブラロード，テキストロード，テキスト修正，テキストセーブ，オブジェクト生成を繰り返すのは，カセットテープ派にとって，マシン語習得の大きな足かせとなっています。そこでアセンブラとテキストを裏RAMに格納し裏RAMで走らせて，図2のA領域にオブジェクトを生成します。暴走すればリセットし，本プログラムをロードし，再びアセンブラを呼び出せば，驚くべき効率でデバッグできます。

その3) 大量配列データを裏RAMに格納する

もともとBASICの変数は恐しく揮発性の強いもので，RUNした瞬間消えてしまいます。そこでGOTO文を使ってプログラムを開始して変数を守るのですが，これでもリストを書き換えれば消え失せてしまいます。大量データをインプットしたけれどリストを修正したいっていうことありませんか？ そういうとき裏RAMにスペアのデータを格納しておけば安心です。たとえばDIMA(20,100) という配列データを$8000～に格納したければデータの先頭アドレスはVARPTR(A (0,0)),終了アドレスはVARPTR (A (20, 100)) +3で求められるから

```
TRANS VARPTR (A (Ø, Ø)), VARPTR (A
  (2Ø, 1ØØ)) +3, &H8ØØØ⏎
```

これで転送されました。もうこの配列を別の用途に使おうが,リストを修正しようが，格納されたデータは不滅です。再び,裏RAMから配列へデータを呼び戻すには

```
TRANS &H8ØØØ, &H8ØØØ+VARPTR (A (2
Ø, 1ØØ)) +3-VARPTR (A (Ø, Ø)), VARP
TR (A (Ø, Ø))⏎
```

とすればOKです(呼び戻す前にDIM命令で領域を確保するのを忘れずに!)。これらの操作はプログラムに組み込むことができるので配列に限らずあらゆるデータを31Kバイト余分に保存できることになり，大量データを扱うプログラム開発をもくろむ人にと

©1983 K. KOREKATA

ダンプリスト

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
7C00  86 0A B7 02 03 30 8D 00 :09
7C08  1E BF 02 04 30 8D 00 51 :F1
7C10  BF 02 10 86 02 B7 02 08 :1A
7C18  30 8D 00 3C BF 02 09 30 :F3
7C20  8D 00 5C BF 02 15 39 43 :3B
7C28  48 41 49 CE 45 52 41 53 :CB
7C30  C5 4C 4C 49 53 D4 4C 50 :69
7C38  52 49 4E D4 53 4F 55 4E :02
7C40  C4 50 4C 41 D9 55 50 4F :6E
7C48  4B C5 54 52 41 4E D3 55 :6D
7C50  45 58 45 C3 55 43 4C D2 :5B
7C58  4C 50 4F D3 55 50 45 45 :ED
7C60  CB 34 02 BD EC 05 35 02 :E6
7C68  81 F4 27 42 81 F5 27 5E :D9
7C70  81 F6 10 27 00 DD 81 F7 :03
7C78  10 27 00 F9 7E 92 A0 BD :9D
7C80  EC 05 BD 00 D8 27 F5 81 :23
7C88  28 26 F1 BD 00 D2 BD 9A :25
7C90  02 BD 00 D8 81 29 26 E4 :4B
7C98  BD 00 D2 B7 FD 0F E6 84 :BC
7CA0  B6 FD 0F 86 00 FD 00 76 :BB
7CA8  86 02 B7 00 17 39 BD 00 :4C
7CB0  D2 26 03 7E 96 63 BD 9A :C9
7CB8  02 BF 00 4B BD 92 92 BD :AA
7CC0  99 BC BE 00 4B B7 FD 0F :21
7CC8  E7 84 B6 FD 0F 39 BD 00 :23
7CD0  D2 27 E0 BD 9A 02 BF 00 :F1
7CD8  4B BD 92 92 BD 9A 02 AF :34
7CE0  8D 00 AC BD 92 92 BD 9A :71
7CE8  02 AF 8D 00 A4 BD 00 D8 :77
7CF0  26 8A BE 00 4B AC 8D 00 :F2
7CF8  96 22 B8 27 47 AC 8D 00 :17
7D00  90 25 1A F7 FD 0F 10 AE :90
7D08  8D 00 86 A6 80 A7 A0 AC :2C
7D10  8D 00 7C 26 F6 A6 84 A7 :F6
7D18  A4 F6 FD 0F 39 EC 8D 00 :58
7D20  6E B3 00 4B E3 8D 00 69 :45
7D28  1F 01 10 AE 8D 00 60 F7 :C2
7D30  FD 0F A6 A4 A7 84 A6 A2 :C9
7D38  A7 82 AC 8D 00 53 26 F6 :D1
7D40  F6 FD 0F 39 B7 FD 0F E6 :E4
7D48  FF 00 4B E7 9D 00 42 B6 :C6
7D50  FD 0F 39 BD 00 D2 10 27 :0B
7D58  FF 59 BD 9A 02 AF 8D 00 :ED
7D60  32 BD 00 D8 10 26 FF 14 :10
7D68  AE 8D 00 27 F7 FD 0F AD :12
7D70  84 F6 FD 0F 39 BD 00 D2 :4E
7D78  10 26 FF 37 F7 FD 0F 8E :FD
7D80  80 00 86 00 A7 80 8C FC :B5
7D88  00 26 F9 F6 FD 0F 39 48 :A2
7D90  49 4B 41 52 55 00 00 00 :7C
```

っては強力な武器となるでしょう。

その4) BASICプログラムをファイルする

ポケットコンピュータでは複数本のプログラムが入りますが，FM-7ではBASICプログラムが，たったの1本しか入りません。プログラムを比較したり，複数のプログラムを交代で使用したいときにはカセットでは時間がかかってたいへんです。そこで，プログラムを裏RAMに退避しておくことができます。FM-7では現在入っているBASICテキストの開始番地，終了番地がそれぞれ＄33・＄34番地，＄35・＄36番地に格納してありますから，MONコマンドで調べてメモしてください。例としてそれぞれ＄0790，＄6809だったとすると，＄8000～に格納する場合，

TRANS &H79Ø, &H68Ø9, &H8ØØØ⏎

これで裏RAMにファイルされました。ここで2本目のプログラムをロードしたり，リストを大きく書き換えてみたりしても，次の操作で元のプログラムを呼び出すことができます。

TRANS &H8ØØØ, &H8ØØØ＋&H68Ø9－&H79Ø, &H79Ø⏎

この後，MONコマンドで＄35・＄36番地に＄68・＄09を書き込みRUNさせます。

いかがでしょうか？　コンピュータによっては増設RAMは数Kバイトで数万円するものもあります。FM-7にはそれが標準装備されているようなものです。本プログラムを用いてFM-7を粋に使いこなしてください。なお本プログラムに限り，個人で利用されるほか，いかなる目的であろうと一部あるいは全部を無断で使用して差し支えありません。

表1

アドレス	意味	参考
＄203 ＄204，205 ＄210，211	ステートメントの拡張数 キーワードテーブルの先頭アドレス 処理ルーチンの先頭アドレス	中間コードは＄F4 から割り当て
＄208 ＄209，20A ＄215，216	関数の拡張数 キーワードテーブルの先頭アドレス 処理ルーチンの先頭アドレス	＄9288をコールすると関数の引数をフローティングアキュムレータに格納する（かっこのチェックも含む）

プログラムについて

本稿で用いたFM-7のBASIC拡張の手法を解説しましょう。

本誌第3号にてFM-8用の拡張の方法が詳しく紹介されていますが，FM-7ではどうすればいいんだろ？　と悩んだ人も多いと思います。まず，1番簡単な方法は，FM-7に存在するが使われていない命令，BUBINI，BUBW，BUBR，を利用することです。MONコマンドでそれぞれ＄5D4・＄5D5，＄5D7・＄5D8，＄5DA・＄5DB番地に実行番地を書き込むだけで，この3つの命令は拡張命令として働きます。マシン語プログラムをBASICにリンクするときUSR関数やEXEC命令より手軽にできますね。しかしこの方法だと命令の名称は決まっているし，3つまでしか拡張できないので，やはり中間コードを新たに定義し，全く新しい命令を拡張したくなります。しかしFM-7はすでに6命令を拡張してあるので話はややこしいわけで，いろいろ考えられるうちで，**表1**のようにするのがベストと思われます。ただしこの場合，ステートメントの拡張数は自分が拡張したい数に6を足した数です。またキーワードテーブルにはCHAIホ ERASナ LLISヤ LPRINヤ SOUNト PLAル を最初に書いてその後に拡張するキーワードを並べてください。関数の場合も同様で，拡張数には1を加え，キー

ワードテーブルにはLPOモの後にキーワードを並べます（今までの話がぜんぜんわからないという人は，本誌第3号を読み返してください）。こうすればディスクでも中間コードが一致するため，全くコンパチに使えます。FM-7の機能を強化するのは楽しいものです。ぜひやってみてください。それからオールRAMに切り換えする際には同時に音が鳴っていると暴走します。＄EC05をコールしてPSGを停止させるのをお忘れなく。

参考文献：Oh!FM Vol. 3 P. 104～108

ソースリスト

```
0000                 ;*******************************
0000                 ;  ALL RAM ACCESS BASIC VER 1.1
0000                 ;      (C) K.Korekata
0000                 ;*******************************
7C00                         ORG  $7C00          $7C00からプログラム作成
7C00 860A                    LDA  #$A            ステートメントの拡張数(10個)
7C02 B70203                  STA  $203
7C05 308D001E                LEAX CODE,PCR       キーワードテーブルの先頭アドレスをロード
7C09 BF0204                  STX  $204
7C0C 308D0051                LEAX ZIKK,PCR       実行ルーチンのアドレスをロード
7C10 BF0210                  STX  $210
7C13 8602                    LDA  #$2            関数の拡張数(2個)
7C15 B70208                  STA  $208
7C18 308D003C                LEAX FCODE,PCR      キーワードテーブルの先頭アドレスをロード
7C1C BF0209                  STX  $209
7C1F 308D005C                LEAX FZIKK,PCR      実行ルーチンのアドレスをロード
7C23 BF0215                  STX  $215
7C26 39                      RTS
7C27 43484149CE      CODE:   FCC  /CHAIﾎERASﾅ/
     45524153C5
7C31 4C4C4953D4              FCC  /LLISﾄLPRINﾄSOUNﾄ/
     4C5052494E
     D4534F554E
     C4                                          拡張ステートメント
7C41 504C41D955              FCC  /PLAﾙUPOKﾅ/    キーワードテーブル
     504F4BC5
7C4A 5452414ED3              FCC  /TRANﾓUEXEﾃ/
     55455845C3
7C54 55434CD2                FCC  /UCLﾒ/
7C58 4C504FD355      FCODE:  FCC  /LPOﾓUPEEﾋ/     拡張関数キーワードテーブル
     504545CB
7C61 3402            ZIKK:   PSHS A              Aレジスタに中間コードを入れて飛んでくるので，それをスタックにしまっておく
7C63 BDEC05                  JSR  $EC05          PSGの音をストップ
7C66 3502                    PULS A              中間コードをスタックから取り出す
7C68 81F4                    CMPA #$F4           もし，Aレジスタ＝＄F4ならば，UPOKEへブランチ
7C6A 2742                    BEQ  UPOKE
7C6C 81F5                    CMPA #$F5           もし，Aレジスタ＝＄F5ならば，TRANSへブランチ
7C6E 275E                    BEQ  TRANS
7C70 81F6                    CMPA #$F6           もし，Aレジスタ＝＄F6ならば，UEXECへブランチ
7C72 102700DD                LBEQ UEXEC
7C74 81F7                    CMPA #$F7           もし，Aレジスタ＝＄F7ならば，UCLRへブランチ
7C78 102700F9                LBEQ UCLR
7C7C 7E92A0          SYNTA:  JMP  $92A0          Syntax Error ルーチンへジャンプ
7C7F                 ;  UPEEK (ｼｷ)
7C7F BDEC05          FZIKK:  JSR  $EC05          PSGの音をストップ
7C82 BD00D8                  JSR  $D8            テキストから1文字取り出してAレジスタに入れる
7C85 27F5                    BEQ  SYNTA          ステートメントエンドの場合はSyntax Errorへ
7C87 8128                    CMPA #$28           取り出した1文字が“(”でなければSyntax Errorへ
7C89 26F1                    BNE  SYNTA
7C8B BD00D2                  JSR  $D2            テキストポインタを1つ増加させる
7C8E BD9A02                  JSR  $9A02          式の値をXレジスタにセットしてリターンしてくる
7C91 BD00D8                  JSR  $D8            テキストから1文字取り出してAレジスタに入れる
7C94 8129                    CMPA #$29           取り出した1文字が“)”でなければSyntax Errorへ
7C96 26E4                    BNE  SYNTA
7C98 BD00D2                  JSR  $D2            テキストポインタを1の増加させる
7C9B B7FD0F                  STA  $FD0F          オールRAMモード切り換え
7C9E E684                    LDB  ,X             Xレジスタに入っている値のアドレスの内容をBレジスタにセット
7CA0 B6FD0F                  LDA  $FD0F          ROMモード切り換え
7CA3 8600                    LDA  #$00           ＄76番地に＄00が，＄77番地にBレジスタの値が入る
7CA5 FD0076                  STD  $76
7CA8 8602                    LDA  #$2            ＄17に整数型(＝2)をセット
7CAA B70017                  STA  $17
7CAD 39                      RTS                 リターン
7CAE                 ; UPOKE ｶｷｺﾐﾊﾞﾝﾁ,ﾃﾞｰﾀ
7CAE BD00D2          UPOKE:  JSR  $D2            テキストポインタを1つ増加，ステートメントエンドの場合はゼロフラグセット
7CB1 2603                    BNE  NOERR          ステートメントエンドでないときはNOERRへ
7CB3 7E9663          ERR:    JMP  $9663          Illegal Function Callへジャンプ
7CB6 BD9A02          NOERR:  JSR  $9A02          式の値(書き込み番地)をXレジスタにセット
7CB9 BF004B                  STX  $4B            書き込み番地を＄4Bにリザーブ
7CBC BD9292                  JSR  $9292          コンマチェック
7CBF BD99BC                  JSR  $99BC          式の値(データ)をBレジスタにセット
7CC2 BE004B                  LDX  $4B            書き込み番地を＄4BからXレジスタにロード
```

```
7CC5 B7FD0F                  STA   $FD0F          オールRAMモード切り換え
7CC8 E784                    STB   ,X             書き込み番地にデータを書き込む
7CCA B6FD0F                  LDA   $FD0F          ROMモード切り換え
7CCD 39                      RTS                  リターン
7CCE               ; TRANS テンソウカイシ,テンソウEnd,テンソウサキ
7CCE BD00D2        TRANS:    JSR   $D2            テキストホインタを1つ増加させ,ステートメントエンドの場合はゼロフラグセット
7CD1 27E0                    BEQ   ERR            ステートメントエンドのときはIllegal Function Callエラー
7CD3 BD9A02                  JSR   $9A02          式の値（転送開始番地）をXレジスタにセット
7CD6 BF004B                  STX   $4B            転送開始番地を＄4Bにリザーブ
7CD9 BD9292                  JSR   $9292          コンマチェック
7CDC BD9A02                  JSR   $9A02          転送終了番地をXレジスタにセット
7CDF AF8D00AC                STX   TENEND,PCR     ワークエリアにリザーブ
7CE3 BD9292                  JSR   $9292          コンマチェック
7CE6 BD9A02                  JSR   $9A02          転送先先頭番地をXレジスタにセット
7CE9 AF8D00A4                STX   TENSAK,PCR     ワークエリアにリザーブ
7CED BD00D8                  JSR   $D8            ステートメントエンドでないときはSyntax Error
7CF0 268A                    BNE   SYNTA
7CF2 BE004B                  LDX   $4B            転送開始番地と転送終了番地を比べる
7CF5 AC8D0096                CMPX  TENEND,PCR
7CF9 22B8                    BHI   ERR            転送開始番地＞転送終了番地のときはIllegal Function Callエラーへ
7CFB 2747                    BEQ   HITOTU         転送開始番地＝転送終了番地のときはHITOTUルーチンへ
7CFD AC8D0090                CMPX  TENSAK,PCR     転送開始番地＜転送先先頭番地のときはUSIROルーチンへ
7D01 251A                    BCS   USIRO
7D03 F7FD0F                  STB   $FD0F          オールRAMモード切り換え
7D06 10AE8D0086              LDY   TENSAK,PCR
7D0B A680          LOOP:     LDA   ,X+            転送開始番地から転送終了番地－1まで転送する
7D0D A7A0                    STA   ,Y+
7D0F AC8D007C                CMPX  TENEND,PCR
7D13 26F6                    BNE   LOOP
7D15 A684                    LDA   ,X             最後の1個を転送
7D17 A7A4                    STA   ,Y
7D19 F6FD0F                  LDB   $FD0F          ROMモード切り換え
7D1C 39                      RTS                  リターン
7D1D EC8D006E      USIRO:    LDD   TENEND,PCR     D＝転送終了番地－転送開始番地＋転送先先頭番地＝転送先終了番地
7D21 B3004B                  SUBD  $004B
7D24 E38D0069                ADDD  TENSAK,PCR
7D28 1F01                    TFR   D,X            XレジスタにDレジスタの値を入れる
7D2A 10AE8D0060              LDY   TENEND,PCR     転送終了番地をYレジスタにセット
7D2F F7FD0F                  STB   $FD0F          オールRAMモード切り換え
7D32 A6A4                    LDA   ,Y             最初の1個を転送
7D34 A784                    STA   ,X
7D36 A6A2          LOOP2:    LDA   ,-Y            転送終了番地－1から,転送開始番地まで転送する
7D38 A782                    STA   ,-X
7D3A AC8D0053                CMPX  TENSAK,PCR
7D3E 26F6                    BNE   LOOP2
7D40 F6FD0F                  LDB   $FD0F          ROMモード切り換え
7D43 39                      RTS                  リターン
7D44 B7FD0F        HITOTU:STA      $FD0F          オールRAMモード切り換え
7D47 E6FF004B                LDB   [$4B]          ＄4Bと＄4Cに入っているアドレスの内容をBレジスタにロード
7D4B E79D0042                STB   [TENSAK,PCR]   Bレジスタの値をTENSAKに入っているアドレスに書き込む
7D4F B6FD0F                  LDA   $FD0F          ROMモード切り換え
7D52 39                      RTS                  リターン
7D53               ; UEXEC カイシハ゛ンチ
7D53 BD00D2        UEXEC:    JSR   $D2            テキストホインタを1つ増加させ,ステートメントエンドの場合はゼロフラグをセット
7D56 1027FF59                LBEQ  ERR            ステートメントエンドのときは,Illegal Function Call エラーへ
7D5A BD9A02                  JSR   $9A02          式の値（開始番地）をXレジスタにセット
7D5D AF8D0032                STX   EXE,PCR        Xレジスタをワークエリアにリザーブ
7D61 BD00D8                  JSR   $D8            ステートメントエンドでないときはSyntax Error へ
7D64 1026FF14                LBNE  SYNTA
7D68 AE8D0027                LDX   EXE,PCR        開始番地をXレジスタにセット
7D6C F7FD0F                  STB   $FD0F          オールRAMモード切り換え
7D6F AD84                    JSR   ,X             開始番地をサブルーチンコール
7D71 F6FD0F                  LDB   $FD0F          ROMモード切り換え
7D74 39                      RTS                  リターン
7D75               ; UCLR
7D75 BD00D2        UCLR:     JSR   $D2            テキストホインタを1つ増加させる
7D78 1026FF37                LBNE  ERR            ステートメントエンドのときは,Illegal Function Call エラーへ
7D7C F7FD0F                  STB   $FD0F          オールRAM モード切り換え
7D7F 8E8000                  LDX   #$8000         ＄8000～＄FBFFを00でクリアする
7D82 8600                    LDA   #0
7D84 A780          LOOP3:    STA   ,X+
7D86 8CFC00                  CMPX  #$FC00
7D89 26F9                    BNE   LOOP3
7D8B F6FD0F                  LDB   $FD0F          ROMモード切り換え
7D8E 39                      RTS                  リターン
7D8F 4849          TENEND:FCC      /HI/           ワークエリア（フログラムの最後にワークエリアがあるときは,
7D91 4B41          TENSAK:FCC      /KA/           このようになにか値を入れておくことをすすめます。）
7D93 5255          EXE:      FCC   /RU/
```

マニュアルバグ情報

「FM-7 F-BASIC文法書」編

P. 2-7 〔例1〕

DIM A$(6)

A$(0)
A$(1)
A$(2)
A$(3)
A$(4)
A$(5)
A$(6)

P. 3-5 〔説明〕の7行目

・コンマまたは，ハイフォンのみを指定したときは，全て……

（挿入）行番号の指定のないとき，および

P. 3-8 〔例〕

LIST "LPT0:",300,1250

P. 3-10 〔形式〕

LLIST [行番号1][{,-} [行番号2]]

P. 3-71 〔形式〕

RESUME [{NEXT |行番号}]

P. 3-101 〔説明〕の4行目

・トラック番号は（ミニフロッピディスクのとき）0から39まで指定できます。

P. 3-106 〔例〕

画面のスクロール範囲を0行から~~15~~（16）行に設定し，……

P. 3-146 〔説明〕のV1.0, V2.0の場合

(1)V1.0, V2.0の場合

:

	誤	正
PF4	~~CONT(CR)~~	GO␣TO␣
PF5	~~LLIST(CR)~~	CONT(CR)
PF6	~~LOAD(CR)~~	?DATE$, TIME$(CR)
PF7	~~SAVE"~~	LIST "LPT0:"(CR)
PF8	~~?DATE$, TIME$(CR)~~	KEY␣
PF9	~~SCREEN7,7(CR)~~	LOAD␣

:

P. 3-213 コマンド名

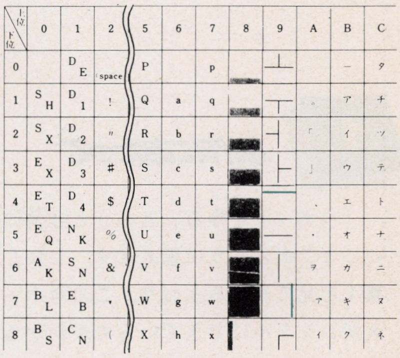
DSKI$(ディスク・アイ・ダラー：disk ~~initialize~~ input $)

P. 3-215 〔説明〕の3行目

……ないときは，キーボードから入力します。キーボードから入力き（したときは），INPUT文と異なり，……

P. 付録-6

付録2キャラクタコード表



虫つくろいのページ

本誌Vol. 5およびVol. . 6に，以下の誤りがありました。訂正し，おわびいたします。Vol.5の訂正については，Vol. 6の「虫つくろいのページ」も合せてご覧ください。なお，色文字部分が訂正事項です。

●Vol. 5　P.27　グラフィックパターンエディタリスト（ただしROM BASICの方のみ）

```
5030 LINEINPUT #1,B$:LINEINPUT#1,B$:GOSUB 5500
5035 IMATR(PATNO,1)=B:IMATR(PATNO,2)=C
```

（追加行）

```
5500 B$=MID$(B$,INSTR(B$,"´")+1,80):P=1
5510 P1=INSTR(P,B$,","):B=VAL(MID$(B$,P,P1-1)):P=P1+1
5520 P1=INSTR(P,B$,","):C=VAL(MID$(B$,P,P1-1)):P=P1+1
5530 P1=INSTR(P,B$,","):MAX=VAL(MID$(B$,P,P-1)):P=P1+1
5540 RETURN
```

●Vol. 6　P.93　タイルペイントプログラムリスト（ただし，FM-7の方のみ）

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
6030  43 BF 02 10 E6 C4 AB C4 :2D
```

●Vol. 6　P.130　3次元迷路プログラムリスト

```
170 LOCATE35,22:PRINT"ﾎﾝﾄｳﾆ?(Y)":KE=1:IF INPUT$(1)="Y" THEN GOTO 280 ELSE 190
```

●Vol. 6　P.51　ブロック崩しリスト4

記事中に完全リロケータブルです，との記載がありますが，一部リロケータブルになっていない所がありました。完全リロケータブルにする場合には，次のデータを追加してください。

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
2FF8  30 8D 0B 51 AF 8D 0B 49 :A9
```

LOADM "FBLOCK", 8

でVol. 6のリストをロードし，MONコマンドで上記の$2FF8～$2FFFを，$3000～$3007として入力してください。

SAVEM "FBLOCK", &H3000, &H3B78, &H3000

で完全リロケータブルのブロック崩しプログラムができあがります。なお，出力は横向きですので，CRTを縦長にして遊んでください。

●Vol. 6　P.99　サンプルプログラムリスト

```
760 TPAINT( 639,0 ),A$, 0
```

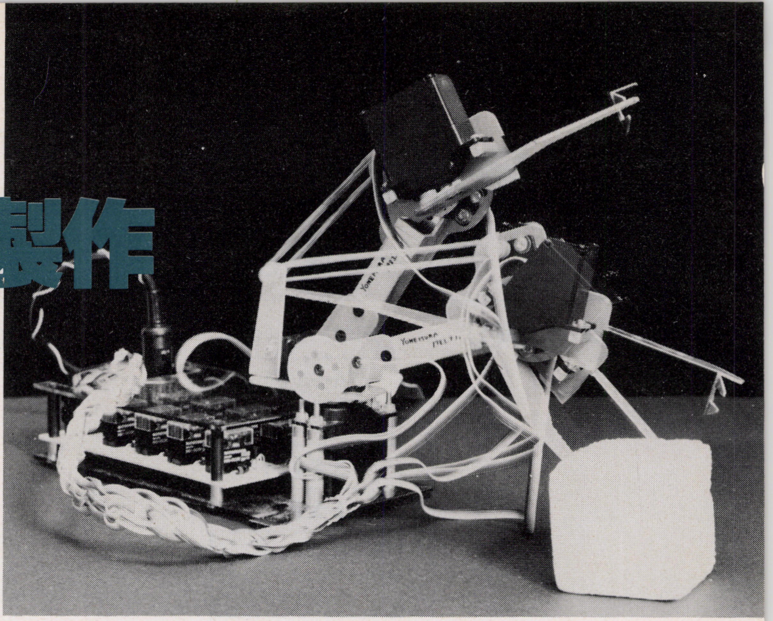
●Vol. 6　P.65　SOUND, PLAY文を使っての音づくり

筆者名が山田康雄氏となっておりますが，山田信満氏の誤りです。

FOR FM-7,8

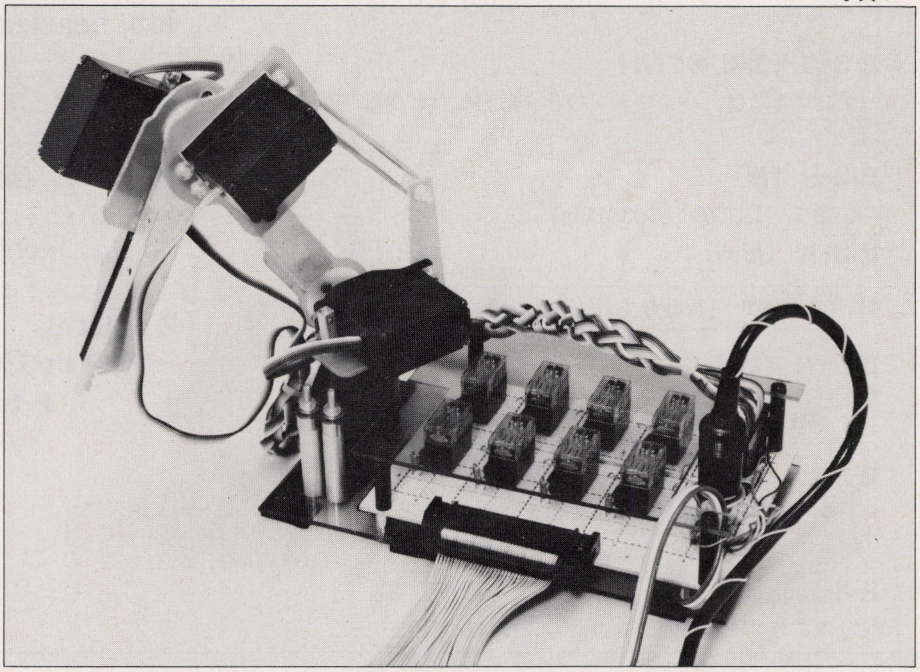
●くれぬなら作ってしまえ《アームロボ》マニピュレータの製作

米村 英明



《アームロボ》プレゼント!!富士通のこの企画にどれほど多くの古くからのFMユーザーが羨望のため息をもらしたことでしょう。私なども V1.05でこそあれEPROM版のふる〜いFM-8を使っているくちで，この広告を無念の涙をこらえてながめたものでした。しかし，皆さん!こんなことでめげてはいけません!!もらえないのならば，自分で作ってしまおうではありませんか!今回は“倍速基板”，“PCG”のときと違い，製作にあたって熟練した腕など必要ありません。根性さえあればできるのです!1983年9月15日以前にFMシリーズを買ってしまった人も，応募しても《アームロボ》をもらえなかった人も，一緒に――なかばヤケ気味で――がんばろうではありませんか!!

▼写真1



1 製作にあたって

「ロボットを作る」というと非常に難しいことであるように思えます。しかし，**写真1**を見てわかるとおり，このロボットは大変シンプルにできているのがわかるでしょう。なぜ，こんなに単純に作れたかというと，実はなんと，あのラジコンに使われるサーボを流用してしまったのです。これを使えば，もう，モータがどうのこうの，ギアボックスがなんのかんの，フィードバックはどうかけようか，などということは一切考える必要がありません。全部ひっくるめて，パックになっているのです。しかもサーボ自体を構造部材として使えるので，まさに良いことずくめです（ただし，値段の問題はありますが…)。今回は安価で入手しやすい，三和電器製作所のSM-394という汎用普及型のサーボを使ってみました。

なお，FM-8用とFM-7用とではインタフェース関係が多少異なりますので，部品集めをする前に，必ず一度，本文を完全に読んでおいてください。

▼表1 本体部品表

サーボ	SM-394(三和)*	4
サーボホーン(JR)		4
グラスファイバ板(2mm)		
または	アルミ板(2mm)	1
アルミ板	(1mm)	1
アルミ角材(15×15mm)		1
L金具		4
皿ビス	M2×6	16
	M3×6	3
ビス	M2×6	2
	M3×6	2
	M3×8	11
	M3×15	2
ナット	M2	18
	M3	18
ワッシャ	3φ	15
カラー	3φ×6	2
スポンジ		少々

* サーボは三和電器製作所東京営業所(台東区浅草橋3-18-1KKKビル)のご好意により，直接そちらに買いに行かれた方には，サーボの中の必要なパーツのみを安価(VR500円，ギヤセット280円，ケース200円，モータ1000円，計1980円程度)にて販売していただけるそうです。

▼表2 インタフェース部品表

リレー HB-1 DC5V	8
2SC945	8
10D1	8
2.2kΩ	8
5kΩ	4
34p フラットケーブル	数m
34p ヘッダ	1
34p コネクタ	2
20p フラットケーブル	50cm
20p ヘッダ	1
20p コネクタ	1
DIN 5p ソケット	1
DIN 5p ケーブル(コネクタ付)	数m
基板	1

▼表3 FM-7の人のための追加部品

Oh!FM 4号のPIA基板	1
Oh!FM 5号のジョイスティック基板	1
DIN 5p ソケット	1

2 部品について

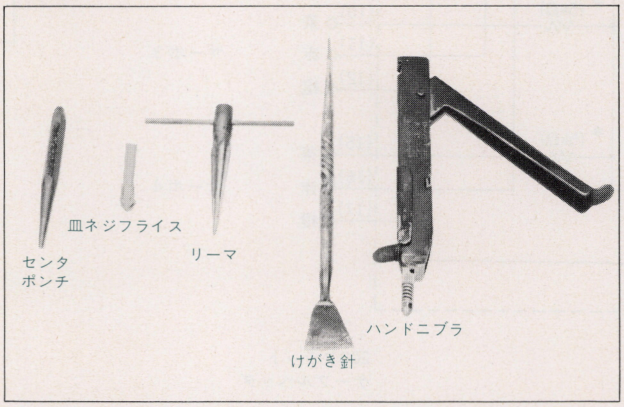
必要な部品を**表1**～**表3**に示します。

SM-394サーボとグラスファイバ板はラジコン店に行けば売っています。サーボは1個4,000円くらい，グラスファイバ板は1枚1,500円くらいですが，秋葉原にあるラジコン店なら2割引くらいで買えます。

サーボホーンは三和の純正のものでは少し薄くて弱いので，JRのものを使いました。これは大小いろいろなサイズのサーボホーンがセットになっていて1パック500円くらいです。フラットケーブル関係については，FMシリーズにちょうど合うものがなかなかないのですが，若松通商で扱っているので，そこでそろえるといいでしょう。なお，フラットケーブルはスダレ状になっているもののほうがよいでしょう。

PIA基板，ジョイスティック基板を除いて，20,000円くらいでそろいます。

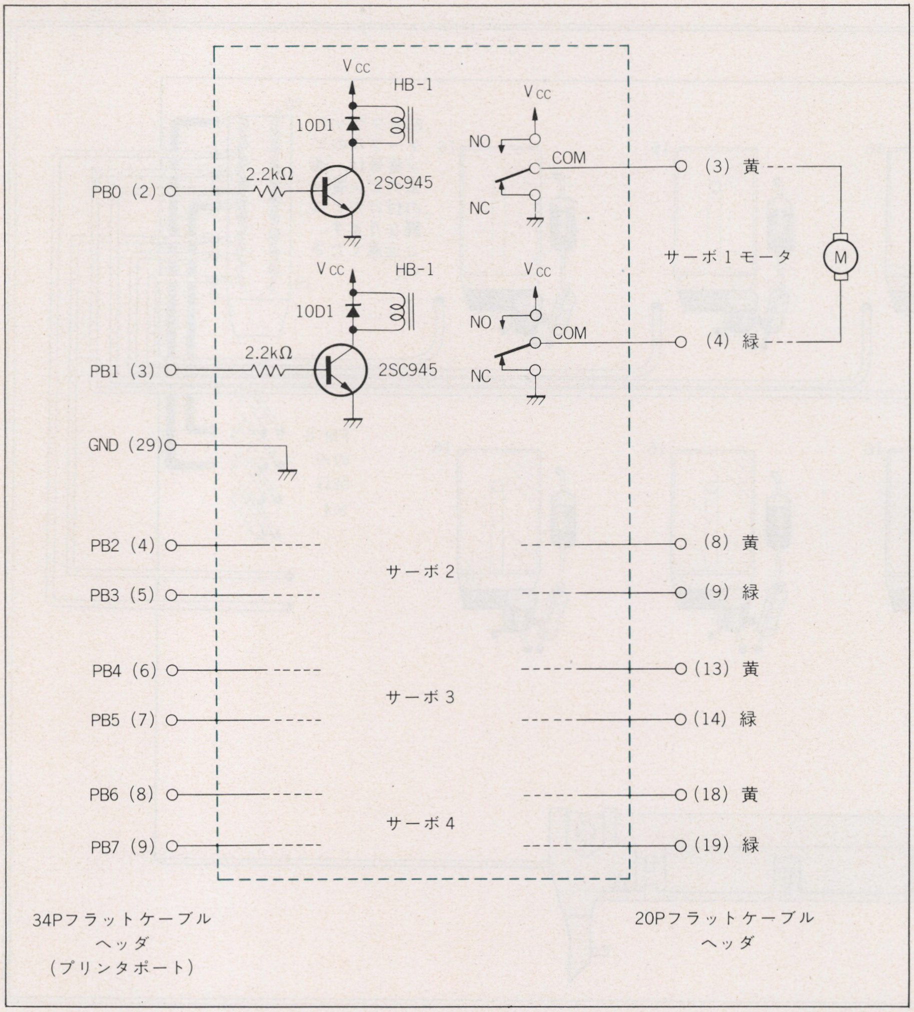
▼写真2



▼表4　工具類

ハンドニブラ
ハンドドリル
または　電動ドリル
センタポンチ
けがき針
ドリル　2mm
　　　　3mm
　　　　6mm
皿ネジフライス
＋ドライバ
ニッパ
ハンダゴテ
ハンダ
リーマ(10φ以上)
金づち
ビニールテープ
やすり
金ノコ
ペンチ

▼図1　出力側回路



3 工具について

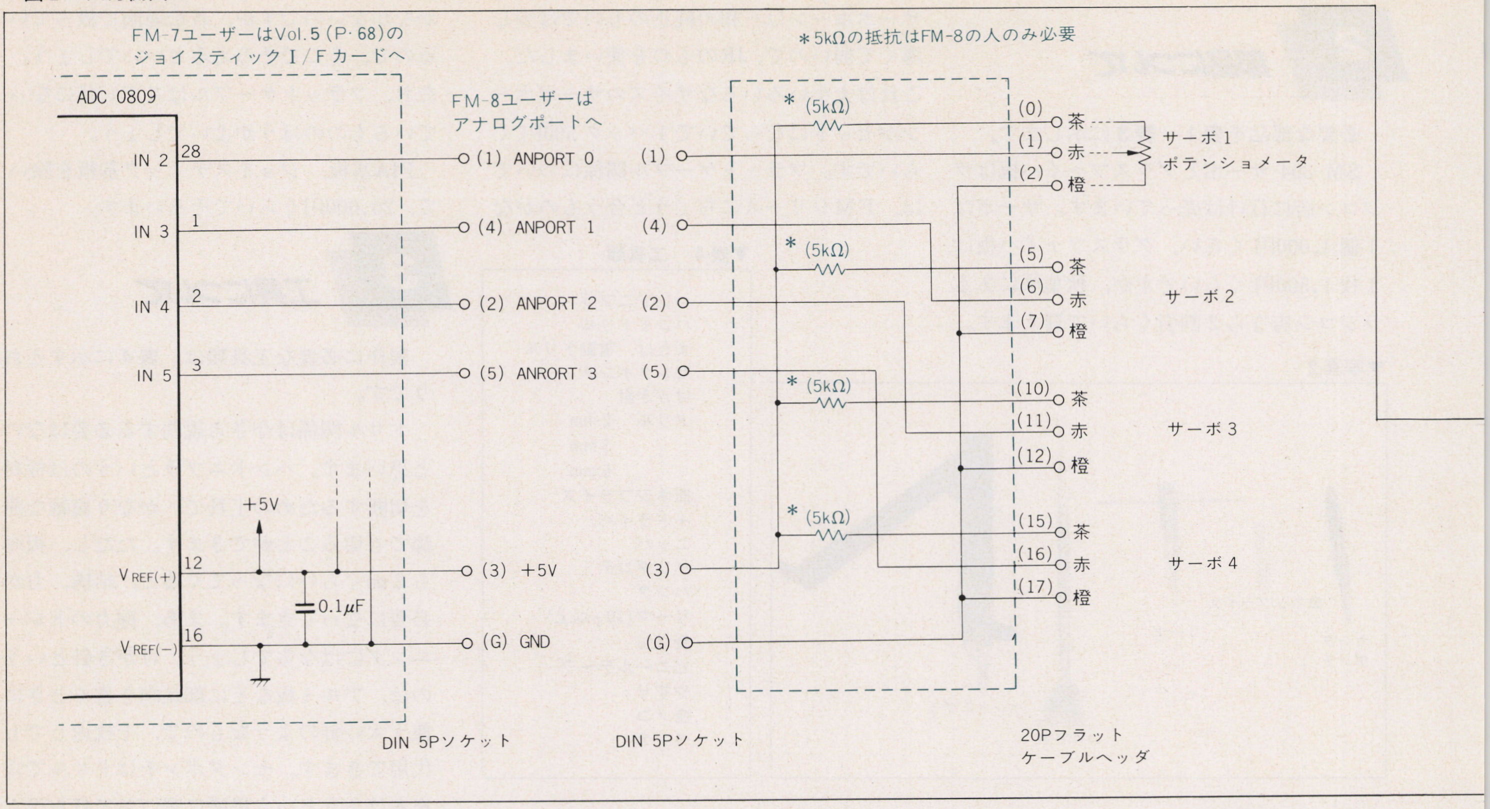
製作に必要な工具類は，**表4**に示すとおりです。

ドリル関係は今さら説明する必要はないと思います。ハンドニブラというのは板材を切断するための工具で，かなり複雑な曲線でも切ることができます。ただし，板厚も2㎜ぐらいになってくると，結構，力が必要になってきます。まあ，握力のトレーニングにはなるでしょう。けがき針というのは，アルミ板などに部品図を書くときに使う太い針のようなもので，千枚通しでも代用できます。センタポンチはドリルで穴をあけようという場所に金づちで打ち込んでくぼみをつくるもので，これを用いると正確な位置に穴をあけることができます。皿ネジフライスは，ドリルであけた穴に皿ネジのための座ぐりを作るためのもので，電動ドリルなどに取り付けられるものが売られています。しかし，小さなハンドドリルの場合，チャックに収まらないことがありますので，その場合には，自分の手で持って穴を削ってください。相手はたかがアルミやプラスチックなので，たいした仕事ではないでしょう。リーマはドリルであけた穴を広げるものです。

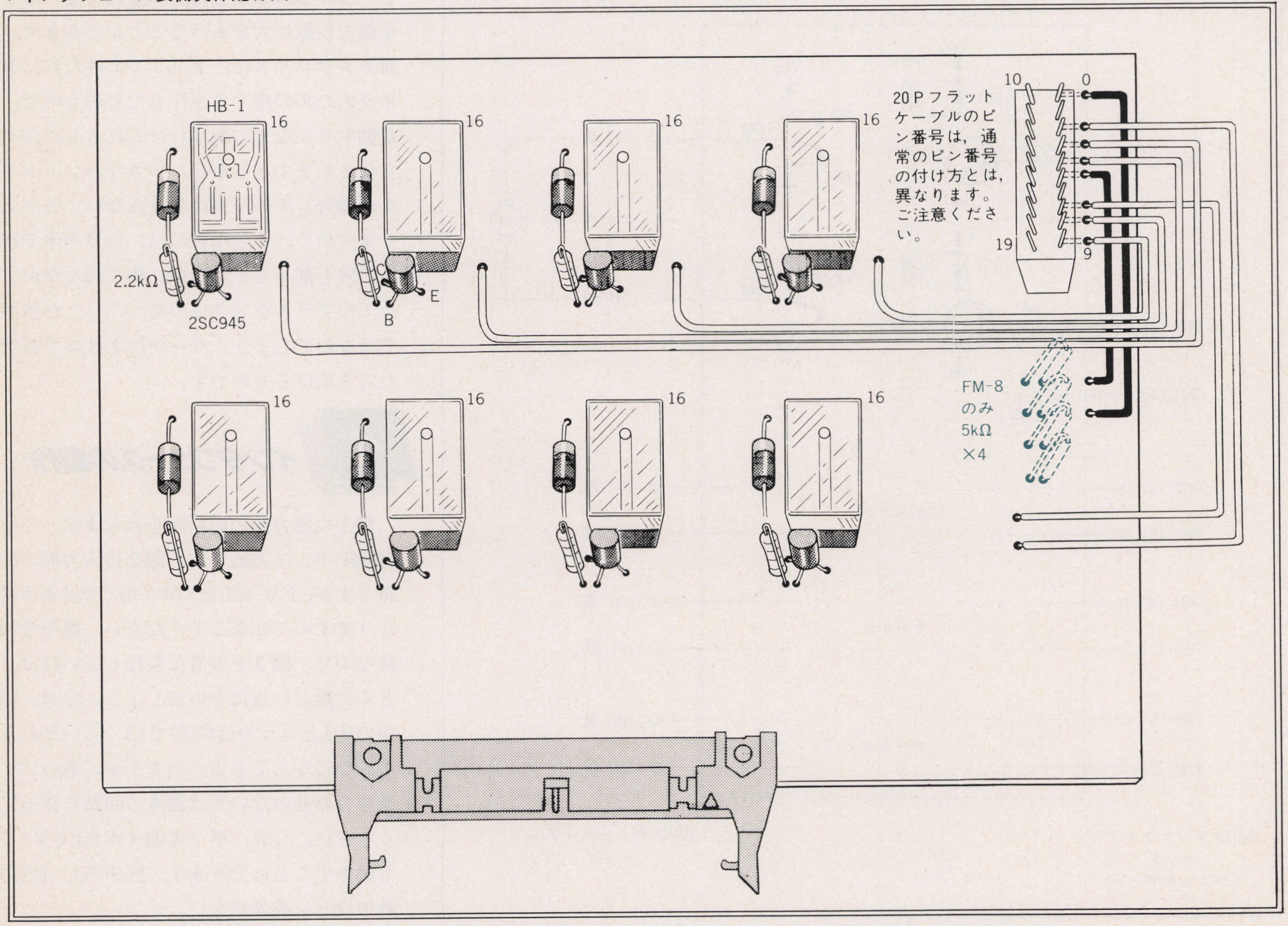
4 インタフェースの製作

図1に出力側の回路図を示します。これはFM-8，7共通です。**図2**は入力側の回路ですが，FM-8用とFM-7用とでは多少異なりますので注意してください。簡単な回路なので，**図3**を参考に製作していけば，とくに難しい点はないでしょう。なお，図中のV_{CC}というのは約5Vで1Aぐらいとれる電源ならなんでもよいのですが，あいにく持ち合わせのない人は**図4**の回路を作ってください。なお，単一電池4本(計6V)でも動かすことはできます。図中の(　)内の数字はピン番号です。

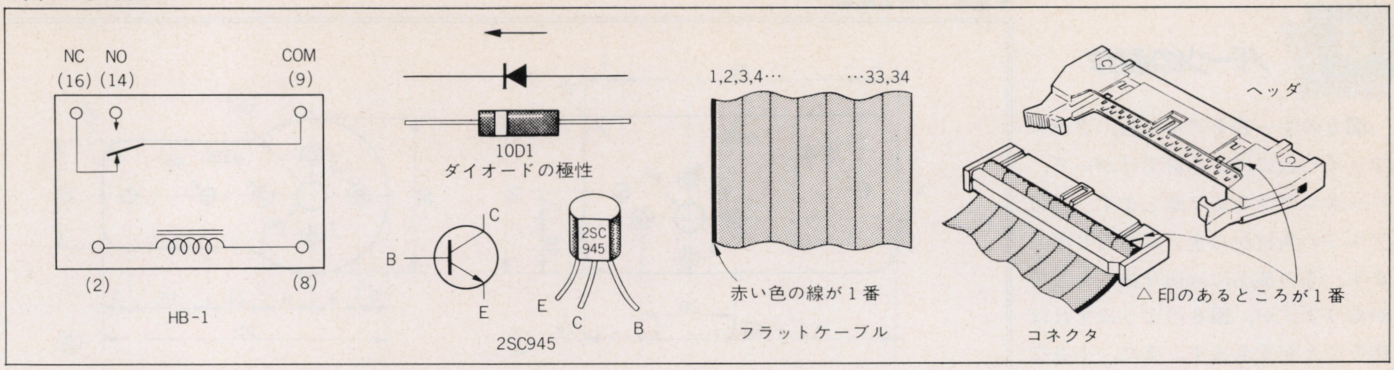
▼図2　入力側回路

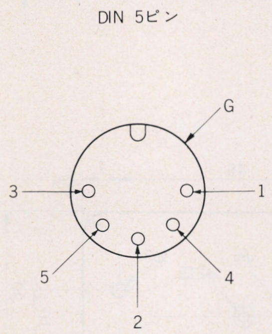


▼インタフェース表側実体配線図



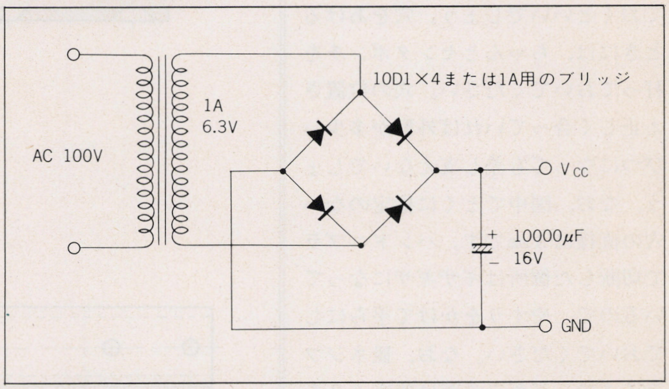
▼図3　参考資料



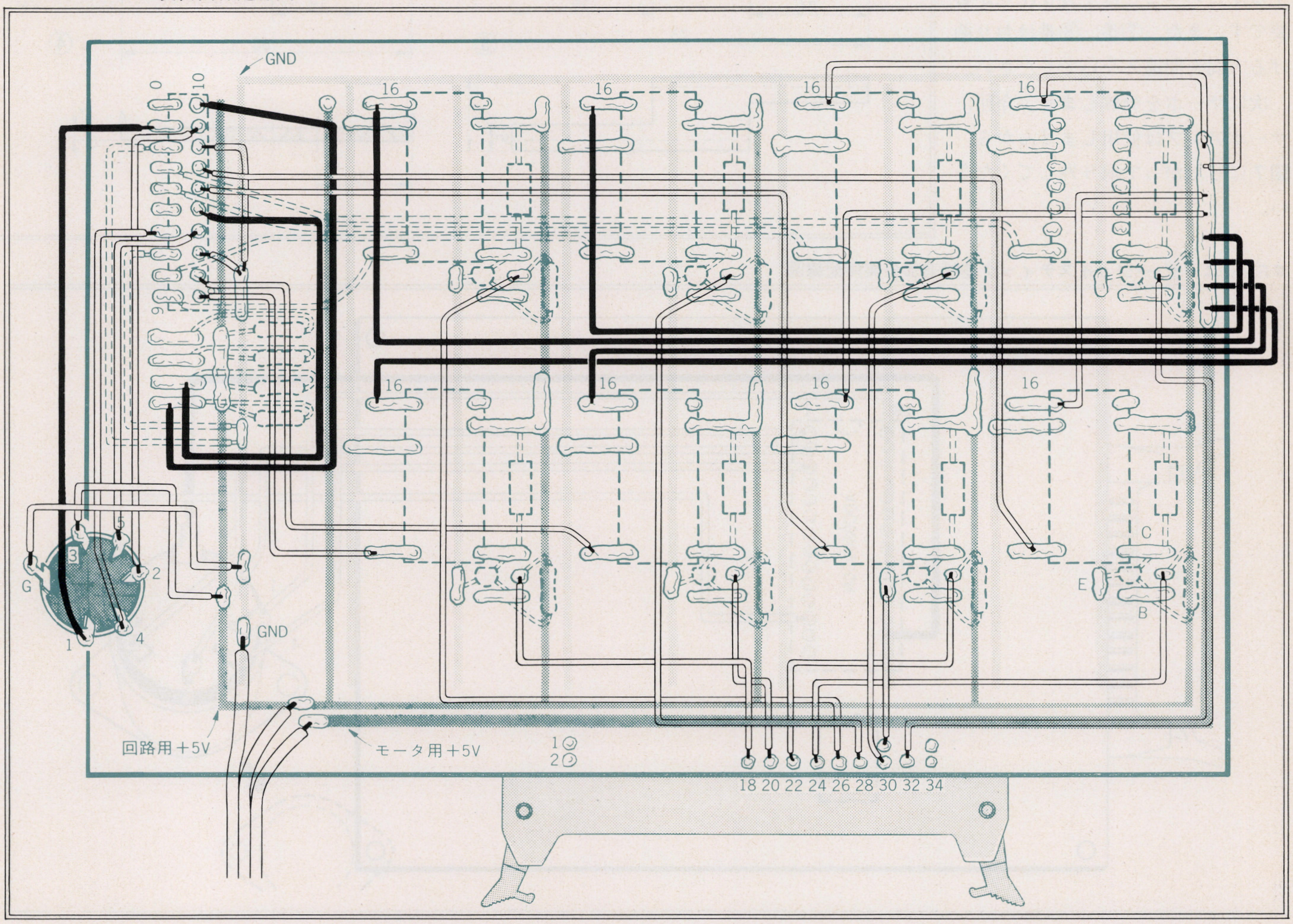


本体コネクタピン番号	信　　号
1	A_0入力
2	A_2入力
3	＋5V
4	A_1入力
5	A_3入力
G	GND

▼図4　電源回路の例



▼インタフェース裏側実体配線図

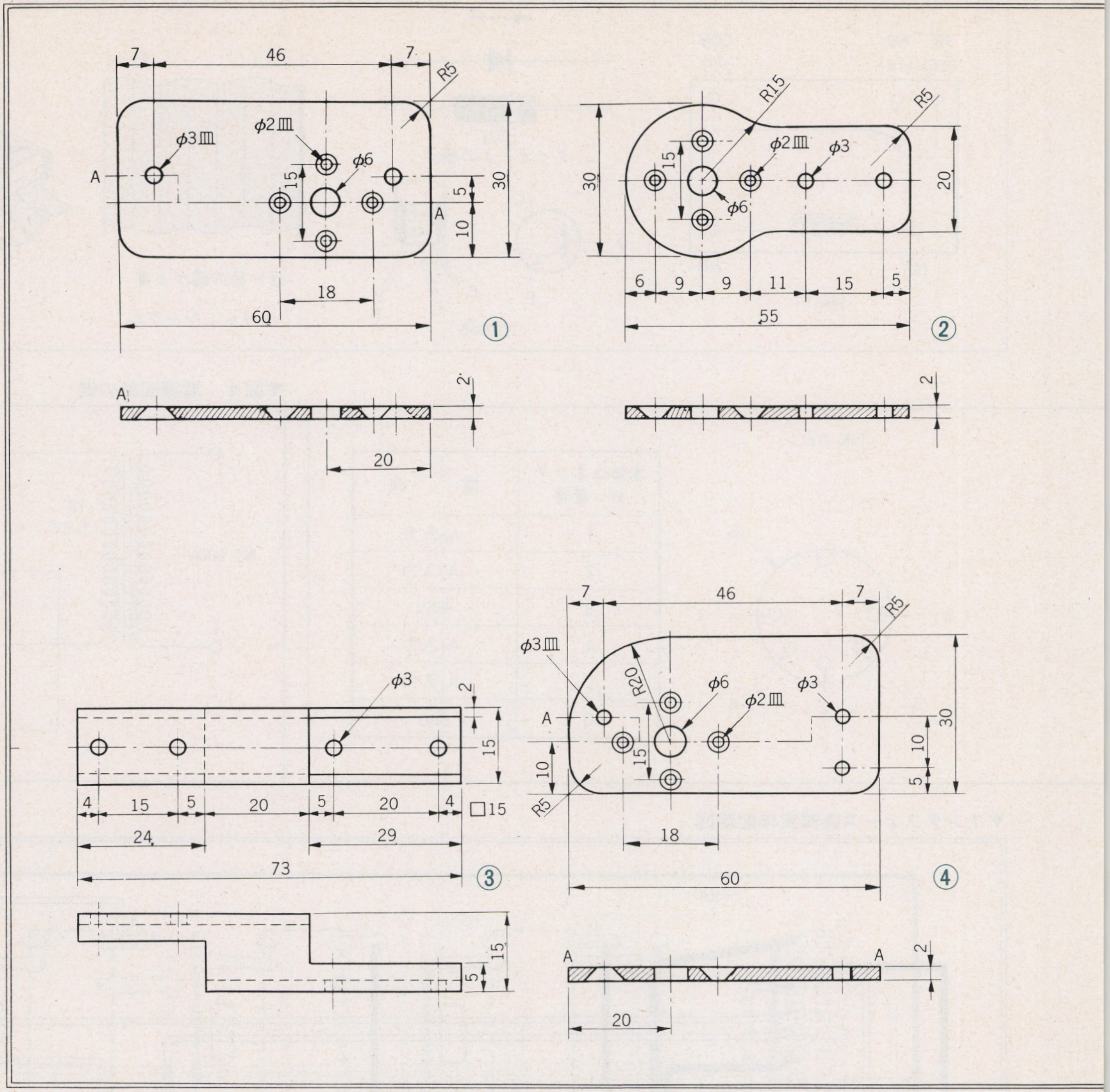


5 パーツの製作

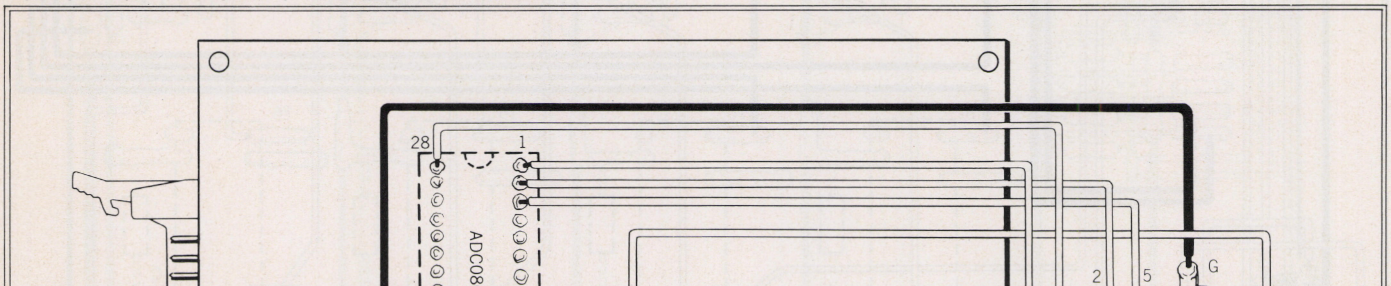
図5の①〜⑧をアルミ板，グラスファイバ板にけがき針でけがいて，ハンドニブラで切り離します。③はアルミの角材から金ノコで切り出します。⑤の部品には切り抜くところがありますが，図6のようにすれば作ることができます。また，小さな部品の穴あけは，切り離す前に行っておくといいでしょう。穴をあけるときには，ちゃんとセンタポンチを打っておいてください。穴の位置さえ正しく合っていれば外形が多少いびつになっても差し支えないでしょう。なお，図中でとくに指定のない穴の直径は3mmです。ハンドニブラで切断した個所はギザギザになっているので，やすりをかけて平らにしておいてください。なお，皿ネジフライスで，どちらの面に皿座ぐりをするかということは十分な注意が必要です。また，⑦番，⑧番はおり曲げる方向を間違えないでください。

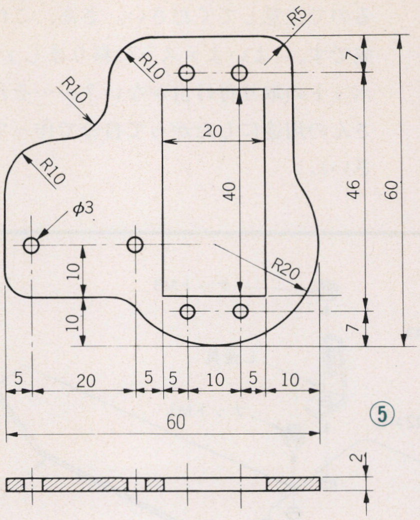
次にサーボを分解します。今回，サーボアンプは使用しませんから，図7にしたがって取り外してください。

▼図5　アーム部品

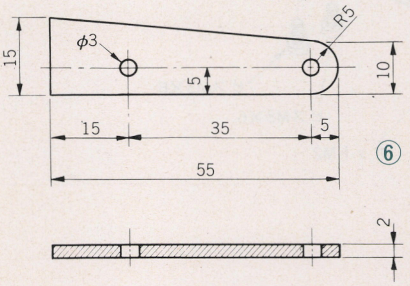


▼FM-7ユーザー・ジョイスティック基板追加分実体配線図



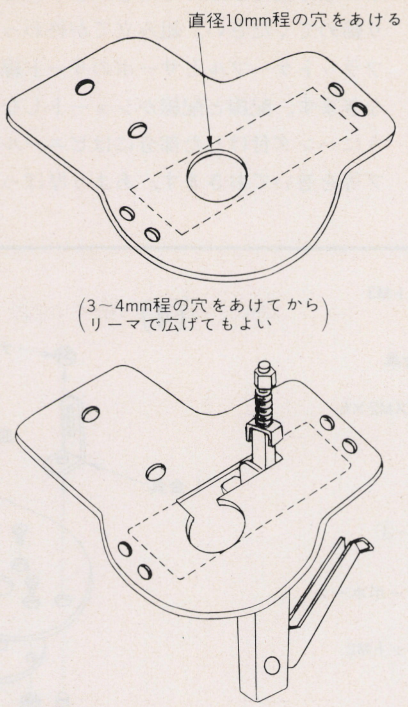


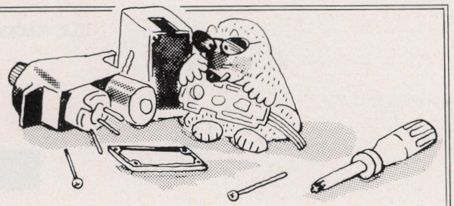
⑤



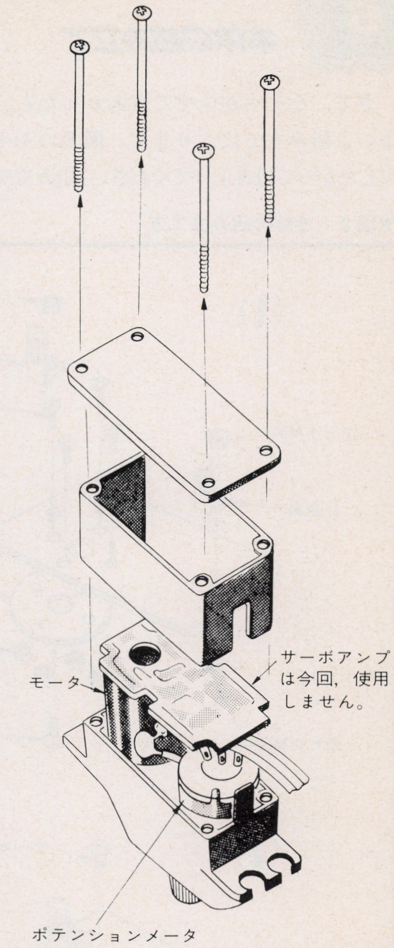
⑥

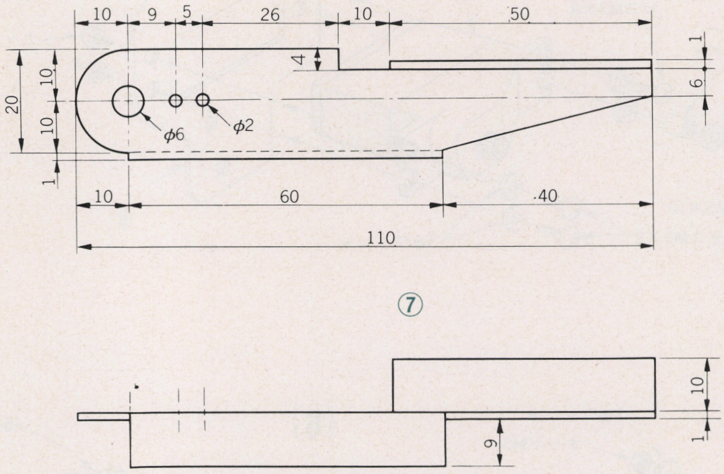
▼図6



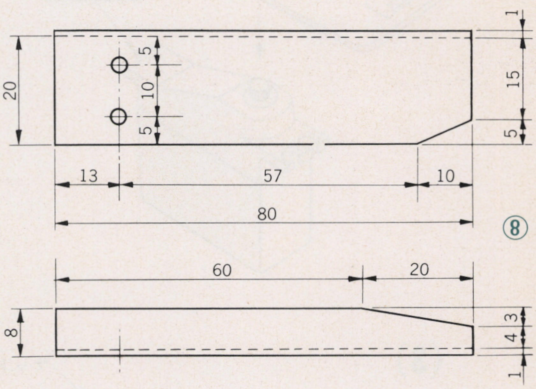


▼図7　SM394分解図

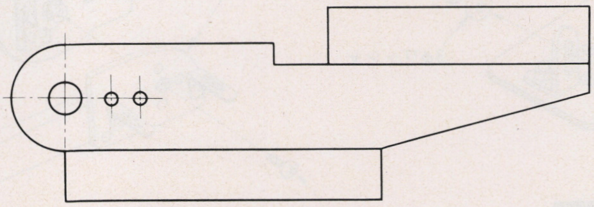




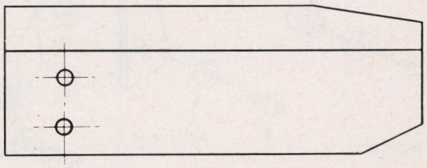
⑦



⑧



⑦の展開図

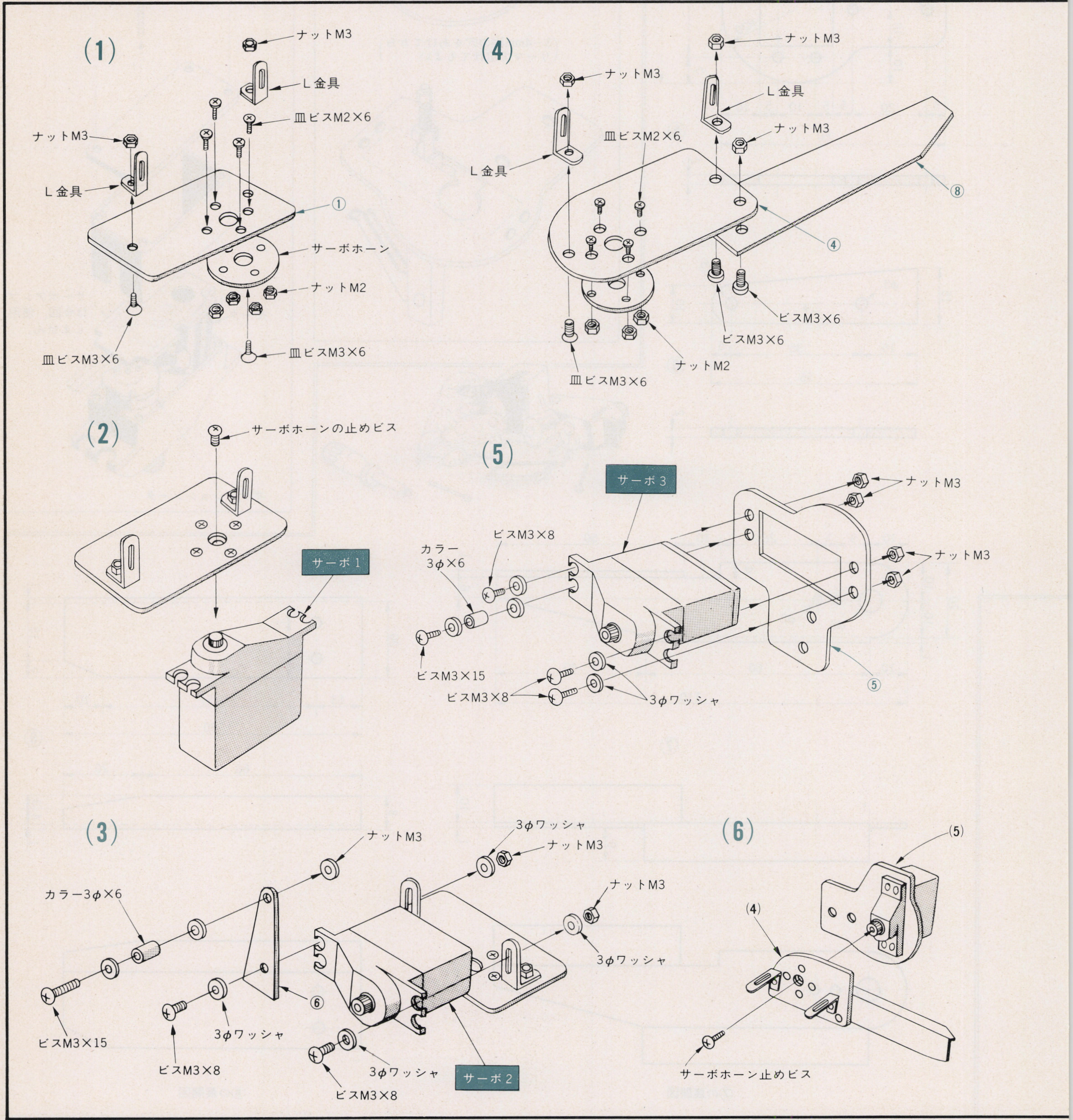


⑧の展開図

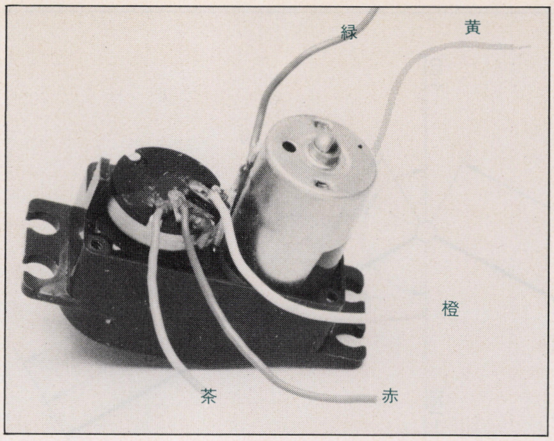
6 本体の組み立て

さて，パーツが全てできあがったら，いよいよ組み立てに入ります。図8の(1)～(11)にしたがって組み上げてください。組み立て時，ネジ類は後でゆるんでこないようにしっかり締めてください。組み立てが終わったら，フラットケーブルとサーボのリード線をつなぎます。配線と配線がショートしないようにハンダ付けした部分にはビニールテープ等を巻いておきます。あまり厚ぼったく巻くとサーボのケース内におさまらなくなるので注意してください。さあ，これで完成です。いよいよテストに移りましょう(ロボットの取り付け台がない？……それは皆さんの用途にしたがって自分で作ってください)。

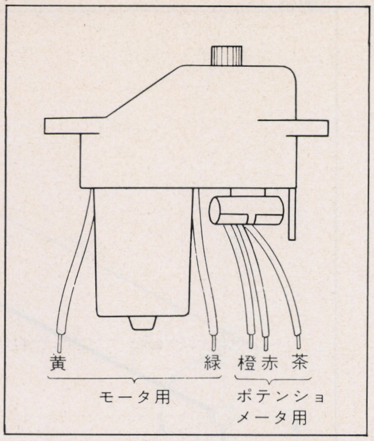
▼図8　本体の組み立て方



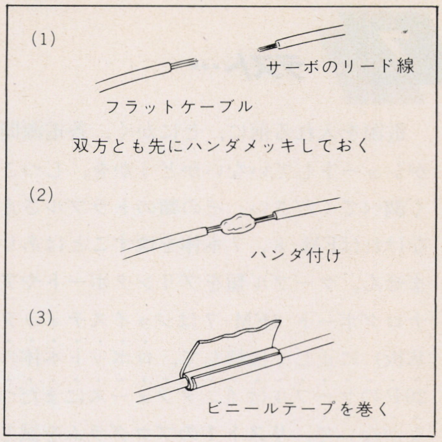
▼写真3



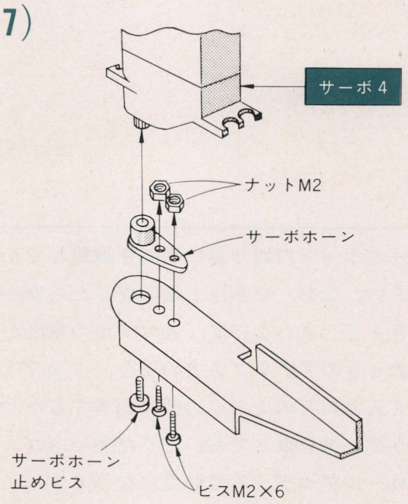
▼図9



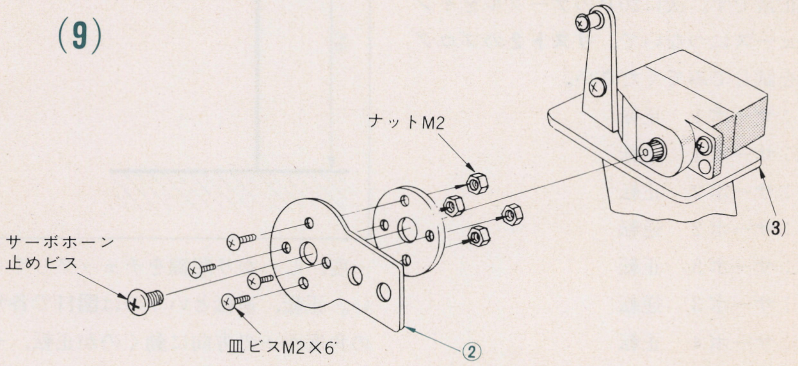
▼図10



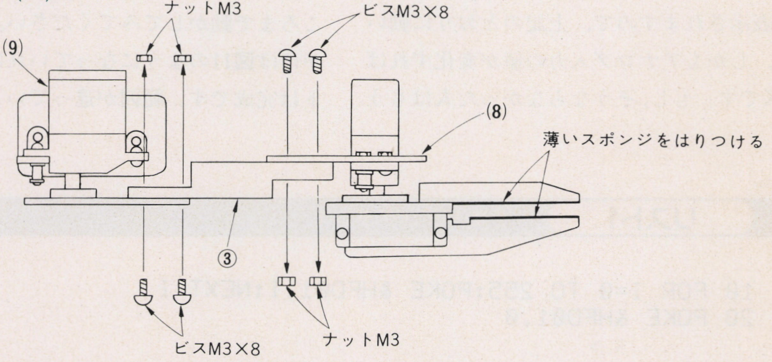
(7)



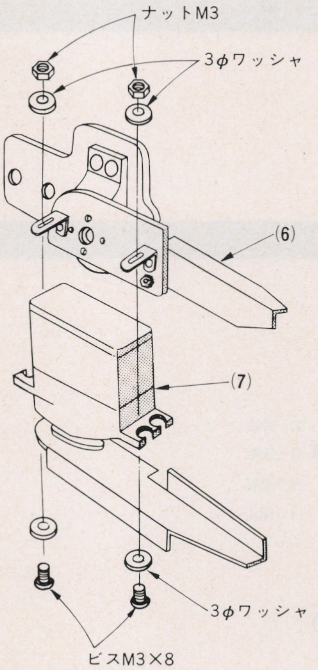
(9)



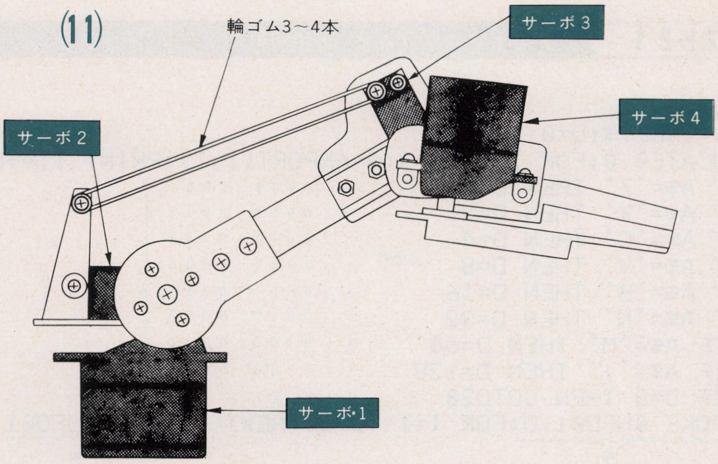
(10)



(8)



(11)



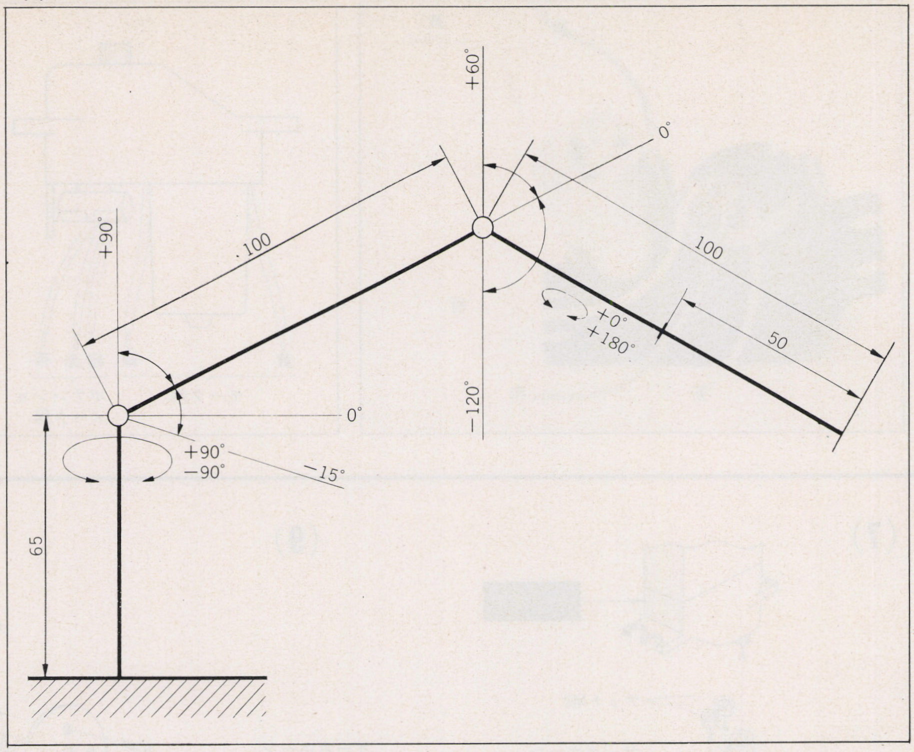
7 テスト…

電源を入れる前に，とにかく，各電源間がショートしていないかどうかを，しつこく調べてください。その類のトラブルさえなければFM-8，7本体を壊すことはありません。ケーブル類をプリンタポートやアナログポート（FM-7はジョイスティック基板）に正しく接続して，ロボット本体の20Pのケーブルはインタフェースにまだつながないで，**リスト1**のプログラムを試してみてください。リレーがすべて正しく動けば正常です。次に20Pのケーブルをインタフェースにつないで，**リスト2**のプログラムを試してみてください。

Z：サーボ1　正転
X：サーボ1　逆転
C：サーボ2　正転
V：サーボ2　逆転
B：サーボ3　正転
N：サーボ3　逆転
M：サーボ4　正転
，：サーボ4　逆転

このとき，CRT上方にアナログ入力の値が表示されますので，上記のとおりに動いて，しかもアナログ入力の値が変化すればOKです。もし，そうならなかった人はもう一度，しっかり配線をチェックしてください。正転，逆転というのは**図11**で各サーボの角度が＋の方向に動くのが正転，－の方向へ動くのが逆転という意味です。次に，各サーボを正転，逆転とも動けなくなるところまで動かしてみてください。動作範囲がほぼ**図11**のようになっていれば，ロボットは完成です。範囲が違っている場合はサーボホーンの取り付け角度を調整してください。なお，ロボットが完成したら安心しきってしまわないで，各サーボの角度が$\theta°$のときのアナログ入力の値と，1°がアナログ入力のどのくらいの値に相当するかというデータを取っておいてください（コントロールプログラムで必要になります）。

▼図11



リスト1　テストプログラム1　© 1983 H.YONEMURA

```
10 FOR I=0 TO 255:POKE &HFD01,I:NEXT I
20 POKE &HFD01,0
```

リスト2-1　テストプログラム2（FM-8用）

```
10 CLS
20 A$=INKEY$:D=0
30 LOCATE0,0:FOR I=0 TO 3:A=ANPORT(I,1):PRINT I;A:NEXT I
40 IF A$="Z" THEN D=1       → 0000 0001
50 IF A$="X" THEN D=2       → 0000 0010
60 IF A$="C" THEN D=4       → 0000 0100
70 IF A$="V" THEN D=8  2進数 → 0000 1000
80 IF A$="B" THEN D=16      → 0001 0000
90 IF A$="N" THEN D=32      → 0010 0000
100 IF A$="M" THEN D=64     → 0100 0000
110 IF A$="," THEN D=128    → 1000 0000
120 IF D=0 THEN GOTO20
130 POKE &HFD01,D:FOR I=1 TO 50:NEXT I:POKE &HFD01,0:GOTO20
```

XX（サーボ4）　XX（サーボ3）　XX（サーボ2）　XX（サーボ1）

00→停止
01→正転
10→逆転
11→停止

POKE &HFD01,D：動作開始
POKE &HFD01,0：全サーボ停止

リスト2-2 テストプログラム2（FM-7用）

```
10 CLS:GOSUB10000
20 A$=INKEY$:D=0
30 LOCATE0,0:FOR I=0 TO 3:GOSUB20000:PRINT I;A:NEXT I
40 IF A$="Z" THEN D=1
50 IF A$="X" THEN D=2
60 IF A$="C" THEN D=4
70 IF A$="V" THEN D=8
80 IF A$="B" THEN D=16
90 IF A$="N" THEN D=32
100 IF A$="M" THEN D=64
110 IF A$="," THEN D=128
120 IF D=0 THEN GOTO20
130 POKE &HFD01,D:FOR I=1 TO 50:NEXT I:POKE &HFD01,0:GOTO20
10000 POKE &HFDFD,&H00:POKE &HFDFC,&H00:POKE &HFDFD,&H14
10100 POKE &HFDFF,&H00:POKE &HFDFE,&H07:POKE &HFDFF,&H2E
10200 RETURN
20000 POKE &HFDFE,I+2
20100 A=PEEK(&HFDFC)
20200 RETURN
```

（10000～10100行：PIAのイニシャライズ）
（20000～20100行：Aにアナログ入力値を入れてリターン）

8 ソフトウェア…

このロボットをコントロールするプログラムはどんなふうに作ればいいのか，ということは**リスト2**のテストプログラムを見れば一目瞭然だと思います。**リスト3**に簡単なコントロールプログラムの例をあげます。しかし，これが最良のプログラムというわけではありません。皆さんでより良いソフトを開発してください。

リスト3のコントロールプログラムの使い方

リスト3を打ち込んでRUNさせると各サーボの現在の角度とハンドの部分のX，Y，Z座標を表示した後，“CMD:”と表示して入力待ちになります。コマンドの種類は

- M：マニュアル操作でロボットを動かす。
- L：動作パターンをロードする。
- S：動作パターンをセーブする。
- R：動作パターンを実行させる。
- T：複数の動作パターンを連結させて実行する。

などがあります。なお，各コマンドを入力すると“PGM:”と聞いてきますが，このとき1～4までの数を入力します。つまり，これによって5種類のパターンを同時に記憶していられるわけです。それらの動作パターンはMコマンドを実行したときにつくられます。Mコマンドを実行すると“Q■”と聞いてきますので動かしたいサーボの番号（1～4）を入力します。次に“deg”と表示されますので角度を入力します。ただ4番のサーボだけは特別で“0”でOPEN，“1”でSHUTです。そして“Q0”，“deg 0”と入力すればMコマンドは終了します。Tコマンドを実行すると，次々とパターンの番号を聞いてきますが終わりにしたい場合は“5”を入力します。LコマンドとSコマンドは実行させるとパターンの番号を聞いた後に，“FILE”と表示されますので，ファイルネームを入力します。なお，1200行と1210行は自分のロボットに合わせて値を変更してください。Z(n)はサーボの角度がθ°のときアナログ入力の値，R(n)は(アナログ入力の値／実際の角度)です。

リスト3-1 コントロールプログラム（FM-8用）…… FM-7の場合，◎印の行はリスト3-2参照

```
1000 '
1010 '<<< MAIN >>>
1020 '
◎1030 WIDTH40,25:GOSUB1130:GOSUB1390:GOSUB1590
1040 LOCATE0,15:INPUT"CMD:",CMD$
1050 IF CMD$="I" THEN GOSUB1590:' ホーム ホ゜シ゛ション イニシャライス゛
1060 IF CMD$="M" THEN GOSUB1670:' マニュアル ソウサ
1070 IF CMD$="R" THEN GOSUB1790:' ハ゜ターンフ゜ロク゛ラム ノ リヒ゜ート
1080 IF CMD$="S" THEN GOSUB2040:' ハ゜ターンフ゜ロク゛ラム ノ セーフ゛
1090 IF CMD$="L" THEN GOSUB2140:' ハ゜ターンフ゜ロク゛ラム ノ ロート゛
1100 IF CMD$="T" THEN GOSUB2240:' ハ゜ターンフ゜ロク゛ラム ノ レンソ゛ク リヒ゜ート
1110 GOSUB1540:GOTO1040
1120 '
1130 '<<< INIT >>>
```

（1030行：GOSUB1130＝変数，関数のイニシャライズ　GOSUB1390＝画面のイニシャライズ　GOSUB1590＝ロボットをホームポジションへ）
（1050～1100行：コマンド入力）
（1110行：GOSUB1540＝画面クリア）

```
1140 '
1150 KEY6,"POKE&HFD01,0"+CHR$(13)
1160 DIM PGM(4,1,49)
1170 DEF FNRAD(Q)=Q*1.74533E-02
1180 DEF FNDEG(Q)=Q/1.74533E-02
1190 L(1)=-90:H(1)=90:L(2)=-15:H(2)=90:L(3)=-120:H(3)=75:L(4)=-90:H(4)=90
1200 R(1)=1.05556:R(2)=1.05556:R(3)=1.13167:R(4)=1.07222
1210 Z(1)=114:Z(2)=228:Z(3)=96:Z(4)=151
1220 DEF FNQ(I,A)=(Z(I)-A)/R(I)
1230 DEF FND(I)=-Q(I)*R(I)+Z(I)
1240 DEF FNX(X)=460+X
1250 DEF FNY(Y)=190-Y/2
1260 DEF FNZ(Z)=Z/2
1270 FOR I=280 TO 630 STEP10:LINE(I,0)-(I,90),PSET,3:LINE (I,100)-(I,190),PSET,4
:NEXT I
1280 FOR I=0 TO 90 STEP5:LINE(280,I)-(630,I),PSET,3:LINE(280,I+100)-(630,I+100),
PSET,4:NEXT I
1290 LINE(460,0)-(460,190),PSET,7
1300 LINE(280,0)-(630,0),PSET,7
1310 LINE(280,190)-(630,190),PSET,7
1320 LOCATE16,0:PRINT"x";:LOCATE16,23:PRINT"x";:LOCATE29,11:PRINT"z";:LOCATE29,1
2:PRINT"y";
1330 RETURN
1340 '
1350 '<<< POSITION >>>
1360 '
1370 LINE(FNX(X1),FNZ(Z1))-(FNX(X2),FNZ(Z2)),PRESET
1380 LINE(FNX(X1),FNY(Y1))-(FNX(X2),FNY(Y2)),PRESET
◎1390 FOR I=1 TO 4:A=ANPORT(I-1,1):Q(I)=FNQ(I,A):LOCATE0,16+I:PRINT"Q";I;"=";CINT
(Q(I));"    ":Q(I)=FNRAD(Q(I)):NEXT I
1400 S1=SIN(Q(1)):C1=COS(Q(1)):S2=SIN(Q(2)):C2=COS(Q(2))
1410 X1=-15*C1+S1*(100*C2+50*COS(Q(3)+Q(2)))
1420 X2=-15*C1+S1*(100*C2+100*COS(Q(3)+Q(2)))
1430 Y1=65+100*S2+50*SIN(Q(3)+Q(2))
1440 Y2=65+100*S2+100*SIN(Q(3)+Q(2))
1450 Z1=15*S1+C1*(100*C2+50*COS(Q(3)+Q(2)))
1460 Z2=15*S1+C1*(100*C2+100*COS(Q(3)+Q(2)))
1470 LOCATE0,22:PRINT"POSITION (x,y,z)"
1480 PRINTUSING"(####,####,####)";X1;Y1;Z1
1490 PRINTUSING"(####,####,####)";X2;Y2;Z2;
1500 LINE(FNX(X1),FNZ(Z1))-(FNX(X2),FNZ(Z2)),PSET,7
1510 LINE(FNX(X1),FNY(Y1))-(FNX(X2),FNY(Y2)),PSET,7
1520 RETURN
1530 '
1540 '<<< CLEAR >>>
1550 '
1560 LINE(0,0)-(15,14)," ",0,BF
1570 RETURN
1580 '
1590 '<<< POS INIT >>>
1600 '
1610 I=2:Q=60:GOSUB1870
1620 I=3:Q=-110:GOSUB1870
1630 I=1:Q=90:GOSUB1870
1640 I=4:Q=1:GOSUB1870
1650 GOSUB1350:RETURN
1660 '
1670 '<<< MANUAL >>>
1680 '
1690 GOSUB1540:LOCATE0,10:INPUT"PGM:",N
1700 IF N<0 OR N>4 THEN 1690
1710 COUNT=0
1720 LOCATE0,11:PRINT"No.";COUNT
1730 INPUT"Q",I:INPUT"deg ",Q:PGM(N,0,COUNT)=I:PGM(N,1,COUNT)=Q
1740 IF I=0 THEN RETURN
1750 GOSUB1870:GOSUB1350
1760 COUNT=COUNT+1:IF COUNT=100 THEN RETURN
1770 GOTO1720
1780 '
```

- 1190: 動作角度の最大値と最小値
- 1200: (アナログ入力→角度)変換用の乗数
- 1220: (アナログ入力→角度)変換
- 1230: (角度→アナログ入力値)変換
- 1240: (X座標→画面のX座標)変換
- 1250: 同Y
- 1260: 同Z
- 1270–1320: 基準座標のワクの表示
- 1370–1380: 以前の座標表示をクリア
- 1390: アナログ入力から各サーボの現在の角度を調べる
- 1400–1460: 各サーボの角度から現在のハンド部分の座標を計算する
- 1470–1510: 各サーボの角度などの表示
- 1560: 前回入力したコマンドの表示を消す
- 1610 (GOSUB1870): ロボットを動かすサブルーチン
- 1610–1650: Iでサーボの番号，Qで角度を指定しロボットをホームポジションへ動かす
- 1650 (GOSUB1350): ロボットの現在の状態の表示
- 1690 (GOSUB1540): コマンド入力部の画面クリア
- 1690 (INPUT): パターンプログラムの番号を入力(0～4)
- 1730: サーボの番号Iと角度Qを入力しそれらを，パターンプログラムに入れる
- 1750 (GOSUB1870): ロボットを動かす
- 1750 (GOSUB1350): ロボットの現在の状態の表示

```
 1790 ´<<< REPEAT >>>
 1800 ´
 1810 GOSUB1540:LOCATE0,10:INPUT"PGM:",N ← パターンプログラムの番号を入力(0～4)
 1820 COUNT=0
 1830 LOCATE0,11:PRINT"No.";COUNT          Iにサーボの番号，Qに角度を入れる    ロボットの現在の状態の表示
 1840 I=PGM(N,0,COUNT):Q=PGM(N,1,COUNT):IF I=0 THEN GOSUB1350:RETURN
 1850 GOSUB1870:COUNT=COUNT+1:GOTO1830
 1860 ´              ロボットを動かす
 1870 ´<<< MOVE >>>
 1880 ´
 1890 IF I=4 THEN 2000 ← サーボ4なら別のサブルーチンへ
 1900 IF Q<L(I) OR Q>H(I) THEN RETURN ← 角度のデータが動作範囲にあるかどうかのチェック
 1910 FOR CD=1 TO 3                   正転方向か，逆転方向か
◎1920 A=ANPORT(I-1,1):SG=SGN(FNQ(I,A)-Q):OB=-Q*R(I)+Z(I) ← 角度をアナログ入力値と比較できるように変換する
 1930 IF A>OB THEN C=2^((I-1)*2) ELSE C=(2^((I-1)*2))*2
 1940 IF A=OB THEN RETURN ← 現在の値が目標値ならリターン
 1950 IF ABS(A-OB)<=30 THEN 1980 ← 現在の値と目標値の差が30以内なら低速移動ルーチンへ
◎1960 POKE &HFD01,C:A=ANPORT(I-1,1):IF SG*(OB-A)<=30 THEN POKE &HFD01,0 ELSE 1960 ← 高速移動ルーチン
 1970 POKE &HFD01,C:FOR J=1 TO 1:NEXT J:POKE &HFD01,0                   } 低速移動ルーチン
◎1980 A=ANPORT(I-1,1):IF SG*(OB-A)<=2 THEN NEXT CD ELSE 1970            }
 1990 RETURN
 2000 IF Q=1 THEN 2020
◎2010 POKE &HFD01,64:A=ANPORT(I-1,1):IF A<80 THEN POKE &HFD01,0:RETURN ELSE 2010 } サーボ4 OPEN
◎2020 POKE &HFD01,128:FOR J=1 TO 30:NEXT J:PA=A:A=ANPORT(I-1,1):IF PA=A THEN POKE
  &HFD01,0:RETURN ELSE 2020                                                     } サーボ4 SHUT
 2030 ´
 2040 ´<<< SAVE >>>
 2050 ´
 2060 GOSUB1540:LOCATE0,10:INPUT"PGM:",N          }
 2070 LINEINPUT"FILE ",FI$                          }
 2080 OPEN"O",#2,FI$                                }
 2090 COUNT=0                                       } パターンプログラムのセーブ
 2100 PRINT#2,PGM(N,0,COUNT);PGM(N,1,COUNT)         }
 2110 IF PGM(N,0,COUNT)=0 THEN CLOSE#2:RETURN       }
 2120 COUNT=COUNT+1:GOTO2100                        }
 2130 ´
 2140 ´<<< LOAD >>>
 2150 ´
 2160 GOSUB1540:LOCATE0,10:INPUT"PGM:",N          }
 2170 LINEINPUT"FILE ",FI$                          }
 2180 OPEN"I",#2,FI$                                }
 2190 COUNT=0                                       } パターンプログラムのロード
 2200 INPUT#2,PGM(N,0,COUNT),PGM(N,1,COUNT)         }
 2210 IF PGM(N,0,COUNT)=0 THEN CLOSE#2:RETURN       }
 2220 COUNT=COUNT+1:GOTO2200                        }
 2230 ´
 2240 ´<<< TABLE >>>
 2250 ´
 2260 GOSUB1540:COUNT=0
 2270 LOCATE0,COUNT:INPUT"PGM:",N
 2280 IF N=5 THEN M=COUNT:GOTO2320
 2290 IF N<0 OR N>4 THEN 2270 ← プログラムのナンバーが0≦N≦4でなければ実行開始
 2300 TB(COUNT)=N:COUNT=COUNT+1
 2310 IF COUNT<10 THEN 2270           リピートのサブルーチン
 2320 FOR K=0 TO M-1:N=TB(K):GOSUB1820:LOCATE0,K:PRINT"                ":NEXT K
 2330 RETURN
```

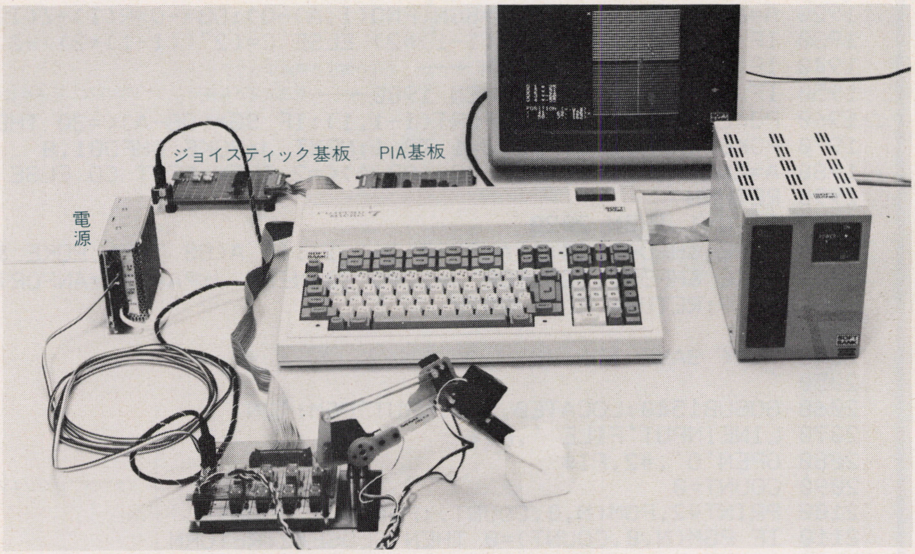
リスト3-2 コントロールプログラム（FM-7用，変更点のみ）

```
1030 WIDTH40,25:GOSUB1130:GOSUB2370:GOSUB1390:GOSUB1590
1390 FOR I=1 TO 4:GOSUB2430:Q(I)=FNQ(I,A):LOCATE0,16+I:PRINT"Q";I;"=";CINT(Q(I))
;"  ":Q(I)=FNRAD(Q(I)):NEXT I
1920 GOSUB2430:SG=SGN(FNQ(I,A)-Q):OB=-Q*R(I)+Z(I)
1960 POKE &HFD01,C:GOSUB2430:IF SG*(OB-A)<=30 THEN POKE &HFD01,0 ELSE 1960
1980 GOSUB2430:IF SG*(OB-A)<=2 THEN NEXT CD ELSE 1970
2010 POKE &HFD01,64:GOSUB2430:IF A<80 THEN POKE &HFD01,0:RETURN ELSE 2010
2020 POKE &HFD01,128:FOR J=1 TO 30:NEXT J:PA=A:GOSUB2430:IF PA=A THEN POKE &HFD0
1,0:RETURN ELSE 2020
```

```
2340 '
2350 '<<< PIA INIT >>>
2360 '
2370 POKE &HFDFD,&H00:POKE &HFDFC,&H00:POKE &HFDFD,&H14
2380 POKE &HFDFF,&H00:POKE &HFDFE,&H07:POKE &HFDFF,&H2E
2390 RETURN
2400 '
2410 '<<< READ ANALOG DATA >>>
2420 '
2430 POKE &HFDFE,I+1
2440 A=PEEK(&HFDFC)
2450 RETURN
```

2370～2380: PIAのイニシャライズ

2430～2450: アナログ入力をAに入れてリターン



9 終わりに

ロボットというものは，意外に簡単に作れるものだ，ということがわかっていただけたかと思います。皆さんも，この作例にとらわれず，自分のオリジナルのロボットを作ってみてはいかがでしょうか。

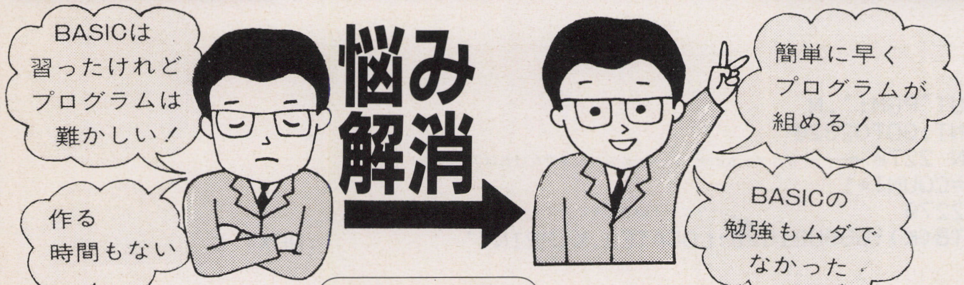
好評発売中

あなたも今日から プログラマー!!

BEST ベスト

FOR BUSINESS EVERY PURPOSE SOFTWARE TOOL

ビジネスユースの為の 簡易ソフトウェアツール!!



適応機種
- 富士通FM-7
- 富士通FM-8
- 富士通FM-11
- NEC PC-8800
- NEC PC-9800

特徴

1. 難しいプログラムをパーツ化!
- 難しいプログラムの部分をパーツ化してフロッピーに格納してありますからお客様は自由に取り出して、つなぎあわせ、必要な変数やデータを書き込むだけで、本格的なビジネスプログラムが作成できます。
- パッケージソフトや、汎用ソフトと違い、内容は全くのハンドメイドですから利用者の思想が充分盛り込め、拡張性があります。

提供媒体 ●フロッピィディスク（5インチ・8インチ）

※上記に関するお問い合わせは企画部 福岡・柘植まで。

日本コンピュータスクール事業協同組合加盟

太陽グループ

理工産業株式会社

〒510 三重県四日市市九の城町4-20

TEL〈0593〉51-1651㈹

- 四日市教室 〒510 三重県四日市市九の城町4-20 ☎(0593)51-1651
- 津教室 〒514 三重県津市本町23-17 ☎(0592)25-6251
- 松阪教室 〒515 三重県松阪市船江町785 ☎(0598)51-4312

F-BASIC中級入門

蓉子の住所録

最終回

菊池　蓉子

はじめに

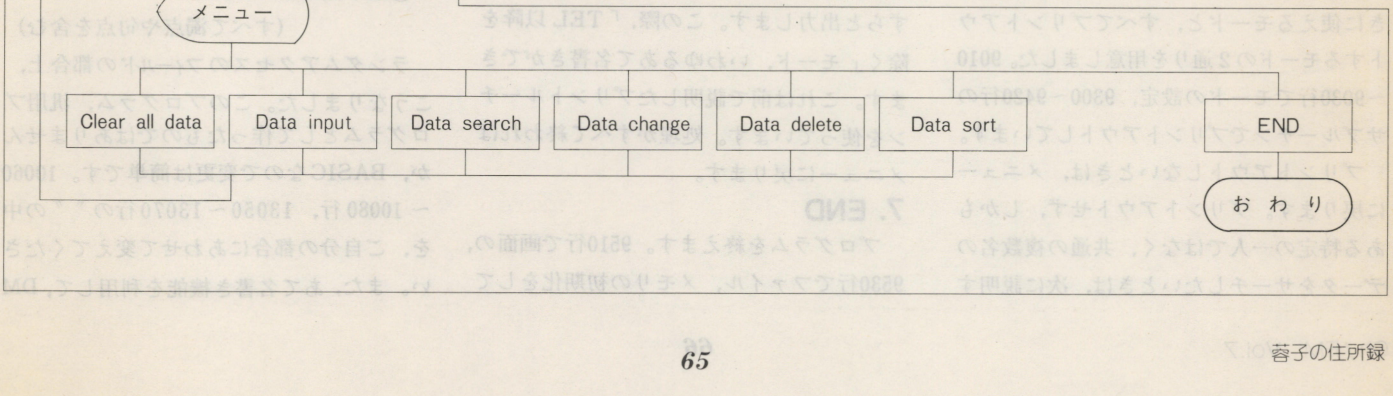
はじめまして，私,「Oh！FM」協力スタッフのニューフェイスです。病気静養中の桑原さんにかわって，私が「F-BASIC中級入門」を引き継ぐことになりました。よろしく。でもちょっと心配，確かに大学でミニコン相手にパスカルぐらいはかじっているけれど，協力スタッフといっても，本当はお茶くみの予定だったのに，プログラム組めるかしら。でも引き受けたからには，女の意地を見せなければ！

さて,「F-BASIC中級入門」最終回は，住所録をつくる予定でしたネ。それでは，さっそく全体的な仕様を考えてみましょう。

★ハード構成

1．FM-7,8,11本体
2．CRT（グリーン，カラーどちらでも可）
3．フロッピディスク（1ドライブで可）
4．プリンタ（なくても可）

図1　ゼネラルフローチャート



★ソフト仕様

1．住所録に必要な項目（郵便番号，氏名，住所，電話番号の4つ）と，誕生日，プレゼント，備考欄を用意。
2．最大1,000件まで。
3．ソートやサーチの高速化のために，MAP方式とランダムアクセスを併用（ただしMAPは氏名，郵便番号の2つ)。
4．ソートは氏名と郵便番号，氏名は頭4文字まで，郵便番号は3文字までは可能で，若い順でも古い順でも可。
5．プリントアウト，すなわちあて名書きも可。
6．漢字用へのバージョンアップも考慮し，余裕をもたせる。

以上の仕様でプログラムを作ってみます。

概略フローチャートと内容

全体的なフローを図1に示します。図のように，大きく7つの仕事に分けました。それぞれの説明をしましょう。

1. Clear all data

これは，今までの住所録を消去して，まったく新しい住所録を作る際に使用します。リストの3000行から3080行までがそうです。ファイルネーム「ADD. DATA」をKILLし，MAPを初期化しています。3060行は，ファイル操作時にエラーを出さないようにするためによく使われる技法です。3070行ではまったく新しいファイルを作るときと，今までのファイルの初期化の両方を考えて，「KILL“ADD. MAP”」を入れてあります。まったく新しいファイルを作る際には，エラー処理ルーチンに飛んでKILLを無視します。そして，メニューに戻ります。

2. Data Input

データを入力するルーチンです。これも1の処理をしてからくれば生じないMAPの問題を4010行で解決しています。また，ファイルのアクセスはMAPよりレコードナンバを得るため，他の処理中，MAPの異常が生じた際，エラールーチンでMAPを再ロードしメニューへ戻ります。10000～10100行のサブルーチンで入力し，4100行でファイルに書き込み，13000行で再出力します。4110～4140行でMAPに必要データを書き込んでいます。続けて入力するときは4050行に戻り，入力しないときはMAPの更新処理をしてメニューへと戻ります。

3. Data Search

ある項目について共通のものを捜し，プリントアウトします。とくに，郵便番号の頭3文字と氏名4文字以内は，MAPを使用して高速です。他の項目や5文字以上の氏名などではディスクをアクセスするので，ちょっと時間がかかってしまいます（とい

っても相手は機械だから，黙々と仕事をしてくれます。私たちはコーヒーブレークにしましょう)。

さて，サーチのフォーマットですが，以下の例を参考にしてください。

例1　アズマ××さんを捜すとき
　アズマ^⏎　と入力
　(^は[^ヘ]をカナモードを解除して押してください)

例2　ホで始まる人を捜すとき
　ホ^⏎　と入力

例3　×××ヨシコさんを捜すとき
　???ヨシコ^⏎　と入力
　ただし?は，わからない文字数分入力してください。

以上の例は他の項目でも同じです。^と?の使い方わかりましたか？ Data Input時に誕生日欄を「1962.01.01」というようなフォーマットで入れておけば，?????01^で1月の誕生日の人をサーチし，プリントアウトしてくれます。また備考欄にお歳暮を送るべき人や年賀状を送るべき人などを，一定のフォーマットを決めて書き込めば，便利に使えます。

プログラムでは5510〜5660行で検索条件を入力します。ただ⏎だけを入力した場合は，^だけとなります。5690行ではMAPの終わりになったら，メニューに戻ります。また5700行では，デリートしたファイルを飛ばす処理をしています。5720〜5770行でMAP中の郵便番号を，5780〜5830行でMAP中の氏名を検索し，見つかるか他の項目で検索するかなどのとき，5850行でファイルをGETしています。そして5860〜5920行でさらに条件に合ったファイルかどうかを詳しく調べています。条件が合えばその最初のファイルを表示し，この条件のものすべてをプリントアウトするかたずねてきます。プリントアウトするときはプリントルーチンへ飛びます。そのルーチンは，あて名書きに使えるモードと，すべてプリントアウトするモードの2通りを用意しました。9010〜9030行でモードの設定，9300〜9420行のサブルーチンでプリントアウトしています。

プリントアウトしないときは，メニューに戻ります。プリントアウトせず，しかもある特定の一人ではなく，共通の複数名のデータをサーチしたいときは，次に説明するData ChangeあるいはData Deleteを使ってください。もちろん，このような目的のときは，本当にチェンジあるいはデリートしないでください。

4. Data Change

ここでは一度入力したファイルの訂正と，一度デリートしたファイル部分への追加の処理をします。まず，訂正すべきファイルをサーチします。もしサーチしたファイルを訂正しないとき，以前に消去したファイルがあれば，そこに追加してよいかどうか聞いてきます。これは，6060〜6140行で処理しています。そして訂正，追加が終われば，6150〜6210行で確認とディスク，メモリへの書き込みを行って，メニューに戻ります。

5. Data Delete

ファイルを消去します。といっても，これはMAP上だけで実際のファイルは消去しません。MAPにデリートマーク(プログラム中では変数DMAKがそれ)を書くだけです。簡単でしょう？ このルーチンもあらかじめ消去したいファイルをサーチします。また，誤って消去しないように2回確認しますのでご心配なく！ ここの処理の要は7090行です。

6. Data Sort

ここでは，ファイルを郵便番号，氏名の頭3文字および4文字でソートします。(キャラクタコードで）若い順か古い順のどちらかが選択できます。8040〜8090行がそうです。ソートのプログラム自体は，桑原さんの「F-BASIC中級入門」中のもので，少しアレンジしただけです。8100〜8270行がそうです。このソートもMAPのみのソートです。

ソートが終わると，CRTかプリンタのどちらかに結果を表示できます。CRTに出力する場合，エスケープキーで止めないと，次々に表示してしまいますので注意してください。プリンタに出力する場合も，ずらずらと出力します。この際，「TEL以降を除く」モード，いわゆるあて名書きができます。これは前で説明したプリントルーチンを使っています。処理がすべて終わればメニューに戻ります。

7. END

プログラムを終えます。9510行で画面の，9530行でファイル，メモリの初期化をしています。このCLOSEは本当は必要ありませんが，ファイルの操作中に起こるディスクファイルの破壊の恐ろしさを知ってもらいたくて，あえて入れておきました(END命令ですべてのファイルはCLOSEされるんです，本当は……)。

8. END以降の行

10000〜10100行……入力用サブルーチン
11000〜11040行……Yes or Noを聞くサブルーチン。⏎だけだとYes
12000〜12020行……入力したストリングに必要な数だけスペースを付加
13000〜13080行……実際のファイルの中身を出力するルーチン
14000〜14070行……サーチのサブルーチン1
15000〜15040行……タイムディレイルーチン
15500〜15530行……MAPロード処理
16000〜16060行……エラー処理ルーチン

実際の使用方法

まずはリストどおりに打ち込んで，セーブしてください。その際ファイルネームは，「ADD. DATA」と「ADD. MAP」の2つを内部で使うので，これらの名前をつけないでください(別のプログラムでこれらの名前があったら，そちらを変更するか，こちらを変更するかしてください)。

あとはRUNさせればわかると思いますが，以下の注意を守ってください。

①郵便番号は最大14桁
②氏名は最大40文字
③住所は最大80文字
④電話番号は最大30桁
⑤誕生日は最大30文字
⑥プレゼントは最大30文字
⑦備考欄は最大30文字
(すべて濁点や句点を含む)

ランダムアクセスのフィールドの都合上，こうなりました。このプログラム，汎用プログラムとして作ったものではありませんが，BASICなので変更は簡単です。10060〜10080行，13050〜13070行の" "の中を，ご自分の都合にあわせて変えてください。また，あて名書き機能を利用して，DM

発行などもできます。

おわりに

「住所録」いかがでしたか？ 一抹の不安を残しながらも，この住所録が多くの人の役に立てばうれしいな，なんて考えています。

ノートの部分や，カナ漢字変換などをバージョンアップできたら，と淡い夢をいだいております。

リスト 蓉子の住所録

©1983 Y. KIKUCHI

```
1000 '♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥
1010 '♥                                                  ♥
1020 '♥   O h ! F M  S e m i D a t a B a s e             ♥
1030 '♥                                                  ♥
1040 '♥ Address Book                                     ♥
1050 '♥ 1.Format of field (254 Bytes)                    ♥
1060 '♥   ┌〒           :14 Bytes for max ┐              ♥
1070 '♥   | NAME        :40               |              ♥
1080 '♥   | ADDRESS     :80       "       |              ♥
1090 '♥   | TELEPHONE   :30       "       |              ♥
1100 '♥   | BIRTHDAY    :30       "       |              ♥
1110 '♥   | PRESENT     :30       "       |              ♥
1120 '♥   └ P.S         :30       "       ┘              ♥
1130 '♥                                                  ♥
1140 '♥ 2.Format of Map (On memory)                      ♥
1150 '♥   ┌ Top  2 Bytes:Number of files         ┐       ♥
1160 '♥   | Next 3 Bytes:Number of 〒             |       ♥
1170 '♥   | Next 4 Bytes:Top 4 Bytes of NAME      |       ♥
1180 '♥   └ Last 2 Bytes:Number for sorting       ┘       ♥
1190 '♥──────────────────────────────────────────────────♥
1200 '♥                                                  ♥
1210 '♥ Programed by Youko.Kikuchi  '83 11/20            ♥
1220 '♥                                                  ♥
1230 '♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥
1240 :
1500 '♥ Procedure INITIALIZE ♥
1510   CLEAR 1000,&H4FFF
1520   DEFINT A-H,J-N,P-W,Y-Z
1530   DEFSTR I,O,X
1540   MTOP=&H5000 : MOF=MTOP+2 : CG=0 : EMAK=254 : DMAK=255
1550   ON ERROR GOTO 16000
1560 :
2000 '♥ Procedure MENU ♥
2010   COLOR 7,0 : WIDTH 40,25 : CONSOLE 0,25,0,0
2020   COLOR 4 : LOCATE  2, 0 : PRINT "* * Oh! FM Original Address Book * *"
2030   LOCATE 2,1 : PRINT "────────────────────────────────────"
2040   COLOR 5 : LOCATE  0, 2 : PRINT "++ M e n u ++"
2050   COLOR 1 : LOCATE  0, 3 : PRINT "─────────────"
2060   COLOR 7 : LOCATE  5, 5 : PRINT "1 : Clear all data"
2070   LOCATE 5, 7 : PRINT "2 : Data Input"
2080   LOCATE 5, 9 : PRINT "3 : Data Search"
2090   LOCATE 5,11 : PRINT "4 : Data Change"
2100   LOCATE 5,13 : PRINT "5 : Data Delete"
2110   LOCATE 5,15 : PRINT "6 : Data Sort"
2120   LOCATE 5,17 : PRINT "7 : E N D"
2130   COLOR 4 : LOCATE  2,21 : INPUT "Input the number of work :";NUM
2140    IF NUM>7 OR NUM<1 THEN 2130
2150   WIDTH 40,25 : CONSOLE 2,23,0,0 : COLOR 4
2160   ON NUM-1 GOTO 4000,5000,6000,7000,8000,9500
2170 :
3000 '♥ Procedure CLEAR ♥
3010   LOCATE 6,0 : PRINT "* *  D A T A  C L E A R  * *"
3020   COLOR 7 : LOCATE 5,1 : PRINT "──────────────────────────";
3030   LOCATE 5,4 : PRINT "ｲﾏﾏﾃﾞﾉ FILE ｦ ｽﾍﾞﾃｹｼﾃｲｲﾃﾞｽｶ ?" : GOSUB 11000 : IF Y=0
 THEN 2000
3040   BEEP : LOCATE 5,7 : PRINT "ﾎﾝﾄｳﾆ ｹｼﾃﾓｲｲﾉﾃﾞｽﾈ ?" : GOSUB 11000 : IF Y=0 TH
EN 2000
3050   LOCATE 5,10 :PRINT "Now Working....."
3060   OPEN "R",#1,"ADD.DATA" : CLOSE #1 : KILL "ADD.DATA"
```

★使用上の注意★

完全にデバッグが済むまでは，1550行および16000行以降のエラールーチンは入れないでください。さもないと，ちゃんとしたエラー処理ができないだけでなく，変なバグ等に悩まされかねません。また，プログラムでは，ファイルオープン命令を使用していますので，ブレークキーで止めた場合は，必ずCLOSE命令をダイレクトで実行してください。ディスクデータの破壊につながりかねませんので。

```
3070  POKE MTOP,0 : POKE MTOP+1,0 : POKE MOF,EMAK : KILL "ADD.MAP" : SAVEM "ADD
.MAP",MTOP,MOF,0
3080  MEND=0 : GOTO 2000
3090 :
4000 '♥ Procedure INPUT ♥
4010   GOSUB 15500
4020  LOCATE 6,0 : PRINT "* * D A T A  I N P U T * *"
4030  COLOR 7 : LOCATE 5,1 : PRINT "──────────────────────────────";
4040  CLS 1 : CHANGE=0
4050   IF MEND=1000 THEN PRINT "Full of Data !!!" : GOTO 4170
4060  AD=PEEK(MTOP)*256+PEEK(MTOP+1) : MEND=AD
4070   GOSUB 10000
4080  PRINT : PRINT "ｺﾚﾃﾞｲｲﾃﾞｽｶ ?" : GOSUB 11000 : IF Y=0 THEN 4160
4090  MEND=MEND+1 : B=MEND : I1=I1+CHR$(B ¥ 256)+CHR$(B MOD 256)+CHR$(EMAK)
4100  OPEN "R",#1,"ADD.DATA" : FIELD #1,254 AS IB : LSET IB=I0 : PUT #1,MEND :G
OSUB 13000 :CLOSE #1
4110  POKE MTOP,MEND ¥ 256 : POKE MTOP+1,MEND MOD 256
4120   FOR J=0 TO 9
4130    POKE MOF+AD*9+J,ASC(MID$(I1,J+1,1))
4140   NEXT J
4150  CHANGE=1
4160  PRINT "ﾂﾂﾞｹﾃ ﾆｭｳﾘｮｸｼﾏｽｶ ?" : GOSUB 11000 : IF Y=1 THEN 4050
4170   IF CHANGE=1 THEN KILL"ADD.MAP" : SAVEM"ADD.MAP",MTOP,MOF+9*MEND,0
4180  GOTO 2000
4190 :
5000 '♥ Procedure SEARCH ♥
5010   GOSUB 15500
5020  LOCATE 6,0 : PRINT "* * D A T A  S E A R C H * *"
5030  COLOR 7 : LOCATE 5,1 : PRINT "──────────────────────────────";
5040   GOSUB 5500
5050  PRINT "ｺﾉｼﾞｮｳｹﾝﾉﾓﾉｦ Print Out ｼﾏｽｶ ?"; : GOSUB 11000 : IF Y=0 THEN 5070
5060   GOSUB 9010
5070  GOTO 2000
5080 :
5500 '♥ Procedure SEARCH Sub 0 ♥
5510  CLS 1 : PRINT "SEARCH ﾉ ｷｰｦﾆｭｳﾘｮｸｼﾃｸﾀﾞｻｲ !"
5520  COLOR 4 : PRINT " (^:ｺﾚｲｺｳﾊﾄﾞｳﾃﾞﾓﾖｲ,?:ｲﾁﾊﾄﾞｳﾃﾞﾓﾖｲ)": COLOR 7
5530  INPUT "〒ﾊﾞﾝｺﾞｳ     ";X : IF X="" THEN X="^"
5540  Y=7 : GOSUB 12000 : X00=X
5550  INPUT "ﾅﾏｴ         ";X : IF X="" THEN X="^"
5560  Y=20 : GOSUB 12000 : X0=X
5570  INPUT "ｼﾞｭｳｼｮ      ";X : IF X="" THEN X="^"
5580  Y=40 : GOSUB 12000 : X1=X
5590  INPUT "Tel.        ";X : IF X="" THEN X="^"
5600  Y=15 : GOSUB 12000 : X2=X
5610  INPUT "ﾀﾝｼﾞｮｳ日    ";X : IF X="" THEN X="^"
5620  Y=15 : GOSUB 12000 : X10=X
5630  INPUT "ﾌﾟﾚｾﾞﾝﾄ     ";X : IF X="" THEN X="^"
5640  Y=15 : GOSUB 12000 : X11=X
5650  INPUT "ﾋﾞｺｳ        ";X : IF X="" THEN X="^"
5660  Y=15 : GOSUB 12000 : X12=X
5670  ST=MOF
5680  CLS 1
5690   IF PEEK(ST)=EMAK THEN PRINT "END" : GOSUB 15000 : EF=0 : RETURN
5700   IF PEEK(ST)=DMAK THEN 5760
5710  X3="^" : X4="?"
5720  S0=1
5730  X5=MID$(X00,S0,1) : IF X5=X3 THEN 5780
5740   IF X5=X4 THEN 5770
5750   IF X5=CHR$(PEEK(ST+S0-1)) THEN 5770
5760  ST=ST+9 : GOTO 5690
5770  S0=S0+1 : IF S0<4 THEN 5730
5780  S0=1
5790  X5=MID$(X0,S0,1) : IF X5=X3 THEN 5840
5800   IF X5=X4 THEN 5830
5810   IF X5=CHR$(PEEK(ST+S0-1+3)) THEN 5830
5820  GOTO 5760
5830  S0=S0+1 : IF S0<5 THEN 5790
5840  B=PEEK(ST+7)*256+PEEK(ST+8)
```

```
5850   OPEN "R",#1,"ADD.DATA" : FIELD #1,254 AS IB: GET #1,B : CLOSE #1
5860   X6=MID$(IB,  1,14) : X7=X00 : GOSUB 14000 : IF S0=0 THEN 5760
5870   X6=MID$(IB, 15,40) : X7=X0  : GOSUB 14000 : IF S0=0 THEN 5760
5880   X6=MID$(IB, 55,80) : X7=X1  : GOSUB 14000 : IF S0=0 THEN 5760
5890   X6=MID$(IB,135,30) : X7=X2  : GOSUB 14000 : IF S0=0 THEN 5760
5900   X6=MID$(IB,165,30) : X7=X10 : GOSUB 14000 : IF S0=0 THEN 5760
5910   X6=MID$(IB,195,30) : X7=X11 : GOSUB 14000 : IF S0=0 THEN 5760
5920   X6=MID$(IB,225,30) : X7=X12 : GOSUB 14000 : IF S0=0 THEN 5760
5930    GOSUB 13000 : EF=-1
5940   RETURN
5950 :
6000 '♥ Procedure CHANGE ♥
6010    GOSUB 15500
6020   LOCATE 6,0 : PRINT "* * D A T A  C H A N G E * *"
6030   COLOR 7 : LOCATE 5,1 : PRINT "────────────────────────────";
6040   CLS 1 : COLOR 4 : PRINT "CHANGE ｽﾙ FILE ｦ SEARCH ｼﾃｸﾀﾞｻｲ !" : GOSUB 15000
 : COLOR 7 : GOSUB 5500
6050   IF EF THEN PRINT "CHANGE ｽﾙﾉﾊ ｺﾚﾃﾞｽｶ "; : GOSUB 11000 ELSE 2000
6060   A=PEEK(MTOP)*256+PEEK(MTOP+1)
6070   F=0
6080    FOR J=0 TO A-1
6090     IF PEEK(MOF+J*9)=DMAK THEN F=-1 : GOTO 6110
6100    NEXT J
6110    IF Y=1 THEN 6140 ELSE IF F THEN CG=PEEK(MOF+J*9+7)*256+PEEK(MOF+J*9+8) E
LSE GOSUB 5760 : GOTO 6050
6120   PRINT "ｱｲﾃｲﾙ ﾄｺﾛﾆ FILE ｦ ｳﾒﾏｽｶ? " : GOSUB 11000
6130    IF Y=0 THEN GOSUB 5760 : GOTO 6050
6140   CLS 1 : GOSUB 10000
6150   PRINT : PRINT "ｺﾚﾃﾞｲｲﾃﾞｽｶ ?" : GOSUB 11000 : IF Y=0 THEN 6040
6160   OPEN "R",#1,"ADD.DATA" : FIELD #1,254 AS IB : LSET IB=I0 : PUT #1,CG : CL
OSE #1
6170   GOSUB 7500
6180    FOR J=0 TO 6
6190     POKE MOF+H*9+J,ASC(MID$(I1,J+1,1))
6200    NEXT J
6210   GOSUB 13000
6220    FOR J=0 TO 500 : NEXT J
6230   GOTO 2000
6240 :
7000 '♥ Procedure DELETE ♥
7010    GOSUB 15500
7020   LOCATE 6,0 : PRINT "* * D A T A  D E L E T E * *"
7030   COLOR 7 : LOCATE 5,1 : PRINT "────────────────────────────";
7040   CLS 1 : COLOR 4 : PRINT "DELETE ｽﾙ FILE ｦ SEARCH ｼﾃｸﾀﾞｻｲ !" : GOSUB 15000
 : COLOR 7: GOSUB 5500
7050   IF EF THEN PRINT "DELETE ｽﾙﾉﾊ ｺﾚﾃﾞｽｶ "; : GOSUB 11000 ELSE 2000
7060    IF Y=0 THEN GOSUB 5760 : GOTO 7050
7070   BEEP : PRINT "ﾎﾝﾄｳﾆ ｹｼﾃﾓｲｲﾉﾃﾞｽﾈ ?" : GOSUB 11000 : IF Y=0 THEN 2000
7080    GOSUB 7500
7090   POKE MOF+H*9,DMAK
7100   GOTO 2000
7110 :
7500 '♥ Procedure CHA. DEL. SUB ♥
7510   H=0
7520    IF PEEK(MOF+H*9+7)*256+PEEK(MOF+H*9+8)=CG THEN 7540
7530   H=H+1 : GOTO 7520
7540   RETURN
7550 :
8000 '♥ Procedure SORT ♥
8010    GOSUB 15500
8020   LOCATE 12,0 : PRINT "* * S O R T * *"
8030   COLOR 7 : LOCATE 9,1 : PRINT "───────────────";
8040   LOCATE 2,3 : PRINT "SORT ﾊ 1:ｵｵｷｲｼﾞｭﾝ(ｳｼﾛｶﾗ)"
8050   LOCATE 9,4 : INPUT "2:ﾁｲｻｲｼﾞｭﾝ(ﾏｴｶﾗ)" ; KIN0
8060   LOCATE 2,6 : PRINT "SORT ﾊ 1:〒 ﾊﾞﾝｺﾞｳ"
8070   LOCATE 9,7 : INPUT "2:ﾅﾏｴ        " ; KIN1
8080    IF KIN0=1 THEN SF0=0 ELSE SF0=-1
8090    IF KIN1=1 THEN SF1=0 ELSE SF1=-1
8100   MAX=PEEK(MTOP)*256+PEEK(MTOP+1)
```

```
8110   DI=0
8120    WHILE 2^DI<MAX
8130     DI=DI+1
8140    WEND
8150   DI=DI-1
8160    WHILE DI>=0
8170     DE=2^DI : DF=1
8180      FOR J=1 TO MAX-DE
8190        IF SF1 THEN MDA=MOF+(J-1)*9+3 : MDB=MOF+(J+DE-1)*9+3 : GOTO 8210
8200       MDA=MOF+(J-1)*9 : MDB=MOF+(J+DE-1)*9
8210       XDA=CHR$(PEEK(MDA))+CHR$(PEEK(MDA+1))+CHR$(PEEK(MDA+2))+CHR$(PEEK(MDA
+3))
8220       XDB=CHR$(PEEK(MDB))+CHR$(PEEK(MDB+1))+CHR$(PEEK(MDB+2))+CHR$(PEEK(MDB
+3))
8230        IF NOT SF1 THEN XDA=LEFT$(XDA,3) : XDB=LEFT$(XDB,3)
8240        IF SF0 THEN IF XDA>XDB THEN GOSUB 8500 : DF=0 : GOTO 8260 ELSE 8260
8250        IF XDA<XDB THEN GOSUB 8500 : DF=0
8260      NEXT J
8270    DI=DI-DF
8280   WEND
8290   COLOR 4 : PRINT " SORT OK !!" : COLOR 7 : GOSUB 15000
8300   CLS 1 : PRINT "ﾂｷﾞﾉﾅｶｶﾗ ｴﾗﾝﾃﾞｸﾀﾞｻｲ"
8310   PRINT : COLOR 4 : INPUT "  1:CRT ﾆ ｼｭﾂﾘｮｸ,2:PRINT OUT    " ; KIN0
8320    IF KIN0=1 THEN 8370
8330   COLOR 7 : PRINT "ﾂｷﾞﾉﾄﾞﾁﾗﾆｼﾏｽｶ "
8340   PRINT : COLOR 4 : INPUT "  1:ｾﾞﾝﾌﾞﾉｺｳﾓｸ 2:TEL ｲｺｳ ｦ ﾉｿﾞｸ" ; KIN1
8350   COLOR 7
8360    IF KIN1=2 THEN PR=0 ELSE PR=-1
8370    FOR J=1 TO MAX
8380     IF PEEK(MOF+(J-1)*9)=DMAK THEN 8450
8390     OPEN "R",#1,"ADD.DATA" : FIELD #1,254 AS IB
8400     B=PEEK(MOF+(J-1)*9+7)*256+PEEK(MOF+(J-1)*9+8)
8410     GET #1,B
8420     CLOSE #1
8430      IF KIN0=2 THEN GOSUB 9300 : GOTO 8450
8440      GOSUB 13000
8450    NEXT J
8460   PRINT " ALL END " : GOSUB 15000
8470   GOTO 2000
8480 :
8500 '♥ Procedure SORT-SUB ♥
8510   MADDA=MOF+(J-1)*9 : MADDB=MOF+(J+DE-1)*9
8520    FOR K=0 TO 8
8530     BANA=PEEK(MADDA+K) : BANB=PEEK(MADDB+K)
8540     POKE MADDA+K,BANB : POKE MADDB+K,BANA
8550    NEXT K
8560   RETURN
8570 :
9000 '♥ Procedure PRINT OUT ♥
9010   CLS 1 : PRINT "ﾂｷﾞﾉﾅｶｶﾗ ｴﾗﾝﾃﾞｸﾀﾞｻｲ"
9020   PRINT : INPUT "1:ｾﾞﾝﾌﾞﾉｺｳﾓｸ 2:TEL ｲｺｳ ｦ ﾉｿﾞｸ " ; KIN
9030    IF KIN=2 THEN PR=0 ELSE PR=-1
9040    GOSUB 9300
9050    IF EF THEN GOSUB 5760 : IF EF THEN 9040
9060   RETURN
9070 :
9300 '♥ Procedure PRINT OUT SUB ♥
9310   OPEN "O",#1,"LPT0:"
9320    IF PR THEN PRINT #1,LEFT$(IB,14) : PRINT #1 : GOTO 9340
9330   PRINT #1,"〒"; : PRINT #1,LEFT$(IB,14) : PRINT #1
9340   PRINT #1,MID$(IB,55,80) : PRINT #1
9350   PRINT #1,MID$(IB,15,40) : IF PR THEN PRINT #1 ELSE PRINT #1,"
 ｻﾏ"
9360    IF PR THEN PRINT #1,MID$(IB,135,30) : PRINT #1
9370    IF PR THEN PRINT #1,MID$(IB,165,30) : PRINT #1
9380    IF PR THEN PRINT #1,MID$(IB,195,30) : PRINT #1
9390    IF PR THEN PRINT #1,MID$(IB,225,30) : PRINT #1
9400   PRINT #1 : PRINT #1
9410   CLOSE #1
```

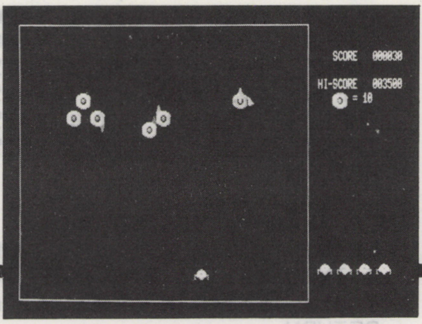
```
9420   RETURN
9430 :
9500 '♥ Procedure END ♥
9510   WIDTH 40,25 : COLOR 7 : PRINT "Program has just finished now!!"
9520   PRINT "See you again. Bye-bye ♥"
9530   CLOSE : CLEAR 300,&H6FFF : END
9540 :
10000 '♥ Procedure INPUT SUB ♥
10010   CLS 1 : I0=""
10020   INPUT "〒          ";X : Y= 7 : GOSUB 12000 : I0=X : I1="" : I1=LEFT$(X,3)
10030   INPUT "ﾅﾏｴ         ";X : Y=20 : GOSUB 12000 : I0=I0+X : I1=I1+LEFT$(X,4)
10040   INPUT "ｼﾞｭｳｼｮ      ";X : Y=40 : GOSUB 12000 : I0=I0+X
10050   INPUT "Tel.        ";X : Y=15 : GOSUB 12000 : I0=I0+X
10060   INPUT "ﾀﾝｼﾞｮｳ日    ";X : Y=15 : GOSUB 12000 : I0=I0+X
10070   INPUT "ﾌﾟﾚｾﾞﾝﾄ     ";X : Y=15 : GOSUB 12000 : I0=I0+X
10080   INPUT "ﾋﾞｺｳ        ";X : Y=15 : GOSUB 12000 : I0=I0+X
10090   RETURN
10100 :
11000 '♥ Procedure YES NO ♥
11010   PRINT "                    OK (Y or N) ?"
11020   I=INPUT$(1) : Y=1 : IF I="Y" OR I="y" OR I="ﾝ" OR I=CHR$(13) THEN RETURN
11030   Y=0 : IF I<>"N" AND I<>"n" AND I<>"ﾐ" THEN 11010
11040   RETURN
11050 :
12000 '♥ Procedure ADD SPACE ♥
12010   X=X+SPACE$(Y*2-LEN(X))
12020   RETURN
12030:
13000 '♥ Procedure PRINT SUB ♥
13010   CLS 1 : COLOR 4 : PRINT "■■ 〒          ■■" : X=LEFT$(IB,14)      : COLOR 7 :
PRINT X : PRINT
13020   COLOR 4 : PRINT "■■ ﾅﾏｴ      ■■" : X=MID$(IB,15,40)  : COLOR 7 : PRINT X
13030   COLOR 4 : PRINT "■■ ｼﾞｭｳｼｮ   ■■" : X=MID$(IB,55,80)  : COLOR 7 : PRINT X
13040   COLOR 4 : PRINT "■■ Tel.     ■■" : X=MID$(IB,135,30) : COLOR 7 : PRINT X
: PRINT
13050   COLOR 4 : PRINT "■■ ﾀﾝｼﾞｮｳ日 ■■" : X=MID$(IB,165,30) : COLOR 7 : PRINT X
13060   COLOR 4 : PRINT "■■ ﾌﾟﾚｾﾞﾝﾄ  ■■" : X=MID$(IB,195,30) : COLOR 7 : PRINT X
13070   COLOR 4 : PRINT "■■ ﾋﾞｺｳ     ■■" : X=RIGHT$(IB,30)   : COLOR 7 : PRINT X
13080   CG=B : RETURN
13090 :
14000 '♥ Procedure SEARCH SUB 1 ♥
14010   S0=1
14020   X5=MID$(X7,S0,1) : IF X5=X3 THEN 14070
14030    IF X5=X4 THEN 14060
14040    IF X5=MID$(X6,S0,1) THEN 14060
14050   S0=0 : GOTO 14070
14060   S0=S0+1 : IF S0<LEN(X6)-1 THEN 14020
14070   RETURN
14080 :
15000 '♥ Procedure DELAY ♥
15010    FOR J=0 TO 500
15020     BEEP 1
15030    NEXT J
15040   BEEP 0 : RETURN
15050 :
15500 '♥ Procedure LOAD ♥
15510    IF PEEK(MTOP)=0 AND PEEK(MTOP+1)=0 AND PEEK(MOF)=0 THEN LOADM "ADD.MAP
"
15520    RETURN
15530 :
16000 '♥ Procedure ERROR ♥
16010    WIDTH 40,25 : BEEP
16020     IF ERR=70 THEN LOADM "ADD.MAP" : RESUME 2000
16030     IF ERR=63 AND ERL=3070 THEN RESUME NEXT
16040    COLOR 2 : PRINT "ｷﾞｮｳﾊﾞﾝｺﾞｳ ";ERL;"ﾆ, ｴﾗｰ ｺｰﾄﾞ ";ERR;" ﾉ ERROR ﾊｯｾｲ!!"
16050    COLOR 4 : PRINT "ｼｮﾘｦ ｲﾁｼﾞ STOP ｼﾏｽ｡"
16060    END
```

FM-7用

ゲームプロセッサ GP-7試用記

矢城　貴志

他人の作ったゲームプログラムで遊ぶだけでは物足りない。といっても，自分でマシン語ゲームを作る自信はない。そんな人のためのゲームプロセッサGP-7を紹介しましょう。



GP-7は，BASICのテキスト上に簡単な命令を書くだけで，オールマシン語のスクロールゲームを作ることができます。たった数行から数十行のプログラムで，インベーダやギャラクシアンのような高速ゲームの類いが作れる！　のです。いわば，ゲームソフトの組み立てキットというべきものです。

GP-7は，A.D.A.オフィス（〒101千代田区神田須田町2-23-1）より，カセットテープ版が4,200円，ディスク版が5,200円で販売されています。ゲームに使うオリジナルキャラクタを作るパターンエディタと，ゲーム中の各キャラクタの動きや各画面の構成を作る簡易コンパイラ言語がセットになっています。

パターンエディタ

24のキャラクタパターンをつくることができます。パターンナンバごとに24×11のます目が表示されるので，カーソル移動キーと0～7の数字でキャラクタを作ります。はじめから自分で作る自信のない人には，UFOなどのパターンが入っていますから，それを手直しすればいいでしょう。

パターンエディタには，このパターンをつくるエディタ機能のほか，つくったパターンをセーブ，ロードする機能がついています。

また，ゲームのタイトルもここでキーボードから入力し，セーブしておきます。

GP-7言語

キャラクタパターンができたら，次はこれらの動きや各画面の構成です。

GP-7言語は，簡単な10種類の命令からできており，初心者にも容易にプログラムが組めるようになっています。それでは，各命令について説明しましょう。

●MVP(m)=〔a，b，c，d，e〕

MVPはキャラクタの動きのパターンを決める命令で，MOVと関連しています。なお，パターンは最大24個定義することができます。

各パラメータの意味は，次のとおりです。

m：パターンの番号となるもので，MOV命令で使われる。

a：動きのパターンで，aの値によって図1のいずれかの動きをする

b：一度画面最下段より消えたとき，再び現れる確率

b＝0……100%　　b＝1……50%

b＝2……25%　　b＝3……15%

b＝4……7.5%

c：画面から消えたインベーダが再び現れるとき，別のキャラクタに変えたいようなときに使う

c＝0……CHIで決めたとおり

c＝1……CHR(0)からCHR(7)のいずれか

c＝2……CHR(2)からCHR(7)のいずれか

c＝3……CHR(4)からCHR(7)のいずれか

c＝4……CHR(6)からCHR(7)のいずれか

d：消えたインベーダが再び現れるときの位置

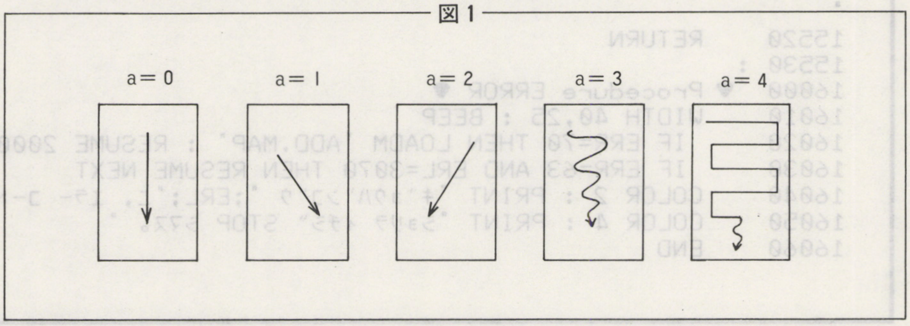
d＝0……INPで決めたとおり（ただしX座標のみ有効）

d＝1……ランダムなX座標で現れる

e：インベーダのキャラクタを交互に表示

e＝0……交互に表示しない

図1



e＝1……交互に表示する（インベーダのキャラクタがCHR(n)のとき，CHR(n)とCHR(n＋1)のキャラクタを交互に表示）

●NUM＝n

ひとつの画面パターン（第1面，第2面といったもの）の中で何個のインベーダを使うか定義します。16個まで表示可能ですが，同じキャラクタでも複数使えば，使用する数だけ勘定に入れなければなりません。また，CHI，MOV，DIF，INP，TAMの各命令ではNUMで定義した数だけデータを入れてやらなければなりません。

●CHR（a）＝n

ひとつの画面で使用するキャラクタパターンを決めます。ただし，1画面につき8つしか定義できません。aはその画面でのキャラクタ番号，nはそのキャラクタを付属のパターンエディタで作成したときにつけた番号です。なお，a×10＋10点がそのキャラクタのインベーダの得点となります。

●CHI〔a_1，a_2，……a_n〕

各インベーダのキャラクタを割り付けます。なお，a_1, a_2, …はCHR(a)で定義したキャラクタ番号です。

●MOV〔m_1，m_2，……m_n〕

各インベーダの動きを割り付けます。m_1，m_2，…はMVP(m)で定義した動作パターンの番号です。

●DIF〔p_1，p_2，……p_n〕

各インベーダが最初から現れているか否かを定義します。現れているときは0，現れていないときは1とします。また1のとき次に現れる確率は，MVPの要素bのとおりです。

●TAM〔q_1，q_2，……q_n〕

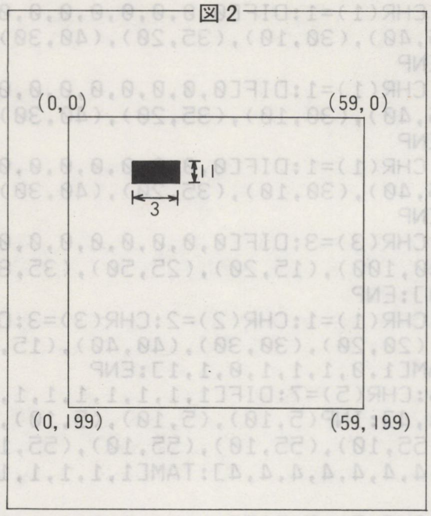
各インベーダがレーザ光線を出すか否かを決めます。レーザ光線を出すなら0，出さないなら1とします。なお，インベーダが一度に発射できるレーザ光線の数には限度があり，先頭のほうで定義されているインベーダが優先されます。

●INP(x_1，y_1)，(x_2，y_2)，……(${x_n}$，${y_n}$)

各インベーダの最初の位置を定義します（ただし$3 \leqq x_i \leqq 57$，$10 \leqq y_i \leqq 176$）。

なお，X座標は8ドット1単位になります（図2参照）。

図2



●PAT（x）：………：ENP

画面パターンの番号を定義します。16個のパターンを定義できますが，PAT(x)とENPの間にNUM，CHR，CHI，MOV，INP，DIF，TAMの7つの定義がなければなりません。

PAT(x)：NUM＝n：CHR(a_1)：CHR(a_2)＝n_2：CHI〔……〕：MOV〔……〕：INP(x_1, y_1)…：DIF〔……〕：TAM〔……〕：ENP

●BGN：x_1—x_2—D—……—x_n：F

ゲームの進行を定義します。パターン〝D〟はすでに決まっており，ドッキング画面です。画面上方を流れていく味方機にタイミングよくBREAKキーを押してドッキングしてください。なお，x_1，x_2…はPAT(x)で定義した画面パターンの番号です。

サンプルプログラム

以上の命令を使って作ったソースリストを**リスト1**に示します。GP-7をお持ちの方はプログラム作りの参考にしてください。また，パターンエディタで作成されたキャラクタのデータは&H3800～&H41FFのメモリ上に保存されます。

リスト2の&H3800～&H3CAFまでを

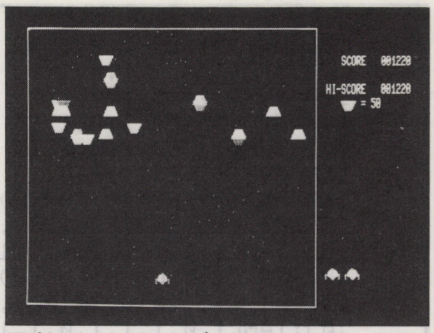


写真1　リスト2（リスト1＋キャラクタデータ）の画面

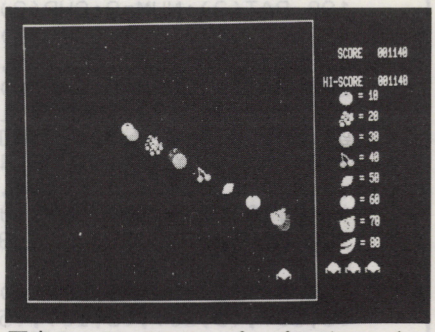


写真2　GP-7のサンプルプログラム〝フルーツパニック〟

打ち込んで，

SAVEM〝ファイル名．#〟，&H3800，&H41FF，Ø⏎

でセーブして，コンパイルしてください。

さて，GP-7をお持ちでない方には，コンパイル後のダンプリストを**リスト2**に示しましたので，リスト2を打ち込んで，

SAVEM〝ファイル名〟，&H3800，&H6FFØ，&H56BØ⏎

でセーブしておきます。

CLEAR300，&H37FF⏎

LOADM〝ファイル名〟，，R⏎

で実行してください。

どうですか。こんなマシン語ゲームが自分で作れたら最高でしょ？　ギャラクシアンやインベーダのようなゲームが簡単に作れます。

なお，GP-7にはサンプルプログラムとして〝フルーツパニック〟というゲームが入っています。

＊　　＊　　＊

最近，こうした〝自分で作るゲームソフト〟の類いがそろそろ出はじめてきました。できあいのゲームに飽きがきた皆さん，このようなソフトでオリジナルゲームを作って楽しんでみませんか。

リスト1

©1983 T.YASHIRO

```
100 MVP(0)=[3,0,0,1,1]
110 MVP(1)=[1,1,0,1,1]
120 MVP(2)=[2,1,0,1,1]
130 MVP(3)=[0,1,0,1,1]
140 MVP(4)=[0,2,0,1,1]
150 MVP(5)=[4,1,0,0,1]
160 PAT(0):NUM=8:CHR(0)=0:CHR(1)=1:DIF[0,0,0,0,0,0,0,0]:CHI[0,0,0,0,0,0,0,0]:INP
(5,10),(15,20),(20,30),(25,40),(30,10),(35,20),(40,30),(45,40):MOV[0,0,0,0,0,0,0,0
,0]:TAM[1,1,1,1,1,1,1,1]:ENP
170 PAT(1):NUM=8:CHR(0)=0:CHR(1)=1:DIF[0,0,0,0,0,0,0,0]:CHI[0,0,0,0,0,0,0,0]:INP
(5,10),(15,20),(20,30),(25,40),(30,10),(35,20),(40,30),(45,40):MOV[0,0,0,0,0,0,0,0
,0]:TAM[1,0,1,1,1,0,1,1]:ENP
180 PAT(2):NUM=8:CHR(0)=0:CHR(1)=1:DIF[0,0,0,0,0,0,0,0]:CHI[0,0,0,0,0,0,0,0]:INP
(5,10),(15,20),(20,30),(25,40),(30,10),(35,20),(40,30),(45,40):MOV[0,0,0,0,0,0,0,0
,0]:TAM[1,0,1,0,1,0,1,0]:ENP
190 PAT(3):NUM=8:CHR(2)=2:CHR(3)=3:DIF[0,0,0,0,0,0,0,0]:CHI[2,2,2,2,2,2,2,2]:INP
(10,10),(20,40),(30,70),(40,100),(15,20),(25,50),(35,80),(45,110):MOV[3,4,3,4,3,
4,3,4]:TAM[1,0,1,1,1,1,1,0]:ENP
200 PAT(4):NUM=8:CHR(0)=0:CHR(1)=1:CHR(2)=2:CHR(3)=3:DIF[0,0,0,0,0,0,0,0]:CHI[0,
0,0,0,2,2,2,2]:INP(10,10),(20,20),(30,30),(40,40),(15,20),(25,50),(35,80),(45,11
0):MOV[0,0,0,0,3,4,3,4]:TAM[1,0,1,1,1,0,1,1]:ENP
210 PAT(6):NUM=16:CHR(4)=6:CHR(5)=7:DIF[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]:CHI[4,4
,4,4,4,4,4,4,4,4,4,4,4,4,4,4]:INP(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10
),(5,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10)
220 MOV[3,3,3,3,3,3,3,3,4,4,4,4,4,4,4,4]:TAM[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]:EN
P
230 PAT(5):NUM=8:CHR(4)=6:CHR(5)=7:DIF[1,1,1,1,1,1,1,1]:CHI[4,4,4,4,4,4,4,4]:INP
(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10):MOV[3,3,3,3,4,4,4,4]:TAM
[1,1,1,1,1,1,1,1]:ENP
240 PAT(7):NUM=8:CHR(0)=0:CHR(1)=1:CHR(4)=6:CHR(5)=7:DIF[1,1,1,1,1,1,1,1]:CHI[0,
0,0,0,4,4,4,4]:INP(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10):MOV[0,
0,0,0,3,4,3,4]:TAM[1,0,1,0,1,1,1,1]:ENP
250 PAT(8):NUM=8:CHR(2)=2:CHR(3)=3:CHR(4)=6:CHR(5)=7:DIF[1,1,1,1,1,1,1,1]:CHI[2,
2,2,2,4,4,4,4]:INP(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10):MOV[3,
4,3,4,3,4,3,4]:TAM[1,0,1,0,1,1,1,1]:ENP
260 PAT(9):NUM=8:CHR(6)=4:CHR(7)=5:DIF[0,0,0,0,0,0,0,0]:CHI[6,6,6,6,6,6,6,6]:INP
(30,70),(20,50),(20,30),(30,10),(40,10),(50,30),(50,50),(40,70):MOV[5,5,5,5,5,5,
5,5]:TAM[0,0,0,0,0,0,0,0]:ENP
270 PAT(10):NUM=8:CHR(6)=10:CHR(7)=11:DIF[0,0,0,0,0,0,0,0]:CHI[6,6,6,6,6,6,6,6]:
INP(30,70),(20,50),(20,30),(30,10),(40,10),(50,30),(50,50),(40,70):MOV[1,2,1,2,1
,2,1,2]:TAM[1,0,1,1,1,0,1]:ENP
280 PAT(11):NUM=8:CHR(6)=10:CHR(7)=11:DIF[0,0,0,0,0,0,0,0]:CHI[6,6,6,6,6,6,6,6]:
INP(10,10),(20,40),(30,70),(40,100),(15,20),(25,50),(35,80),(45,110):MOV[2,2,2,2
,2,2,2,2]:TAM[1,0,1,1,1,1,0,1]:ENP
290 PAT(12):NUM=16:CHR(4)=8:CHR(5)=9:DIF[1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1]:CHI[4,
4,4,4,4,4,4,4,4,4,4,4,4,4,4,4]:INP(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,1
0),(5,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10)
300 MOV[3,3,3,3,3,3,3,3,4,4,4,4,4,4,4,4]:TAM[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]:EN
P
310 PAT(13):NUM=16:CHR(0)=0:CHR(1)=1:CHR(2)=2:CHR(3)=3:CHR(4)=4:CHR(5)=5:CHR(6)=
10:CHR(7)=11:INP(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(5,10),(55,10),
(55,10),(55,10),(55,10),(55,10),(55,10),(55,10),(55,10)
320 DIF[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]:CHI[0,0,0,0,2,2,2,2,4,4,4,4,6,6,6,6]
330 MOV[0,0,0,0,3,4,3,4,5,5,5,5,1,2,1,2]:TAM[1,0,1,1,1,0,1,1,0,0,0,0,1,0,1,0]:EN
P
340 BGN:0-3-1-D-5-4-6-D-6-7-8-D-9-2-10-D-11-12-13:FIN
```

リスト2

©1983 A.D.A.& T.YASHIRO

Add	+0	+1	+2	+3	+4	+5	+6	+7	Sum
3800	01	FF	80	1F	FF	F8	7E	AA	:BE
3808	AE	FA	81	EB	EA	00	7A	EA	:62
3810	00	7A	EA	00	7A	FA	81	FA	:53
3818	7A	FF	EA	0A	BF	A8	00	AA	:7E
3820	80	01	FF	80	1F	FF	F8	7E	:94
3828	AA	AE	FA	81	EB	EA	3C	7A	:5E
3830	EA	7E	7A	EA	3C	7A	FA	81	:FD
3838	FA	7A	FF	EA	0A	BF	A8	00	:CE
3840	AA	80	01	FF	80	1F	FF	F8	:C0
3848	7E	AA	AE	FA	81	EB	EA	30	:56
3850	7A	EA	40	7A	EA	00	7A	FA	:7C
3858	81	FA	7A	FF	EA	0A	BF	A8	:4F
3860	00	AA	80	00	18	00	00	2C	:6E
3868	00	00	2C	00	00	6A	00	00	:96
3870	6A	00	00	6A	00	00	6A	00	:3E
3878	00	6A	00	00	2C	00	00	2C	:C2
3880	00	00	18	00	00	18	00	00	:30
3888	2C	00	00	2C	00	00	6A	00	:C2
3890	00	EB	00	01	EB	80	00	EB	:42
3898	00	00	6A	00	00	2C	00	00	:96
38A0	2C	00	00	18	00	00	18	00	:5C
38A8	00	2C	00	00	2C	00	00	6A	:C2
38B0	00	00	EA	00	01	6A	00	00	:55
38B8	6A	00	00	6A	00	00	2C	00	:00
38C0	00	2C	00	00	18	00	01	FF	:44
38C8	80	07	FA	E0	3E	FA	BC	EE	:43
38D0	80	AF	FF	FB	FF	EE	F3	AA	:B3
38D8	FF	FF	FF	FF	F2	FA	2F	C2	:D9
38E0	A8	07	C2	A0	00	C2	80	01	:54
38E8	FF	80	07	FA	E0	3E	FA	BC	:54
38F0	EE	80	AF	FF	FB	FF	EE	F3	:F7
38F8	AA	FF	FF	FF	FF	F2	FA	2F	:C1

3900	FE	A8	07	FE	A0	00	FE	80	:C9
3908	01	FF	80	07	FA	E0	3E	FA	:99
3910	BC	EE	80	AF	FF	FB	FF	EE	:C0
3918	F3	AA	FF	FF	FF	FF	F2	FA	:85
3920	2F	FA	A8	07	E2	A0	00	C2	:1C
3928	80	01	FF	80	07	FA	E0	3E	:1F
3930	FA	BC	EE	FA	AF	FF	FB	FF	:46
3938	00	F3	80	FF	81	FF	FF	80	:71
3940	FA	2F	C2	A8	07	80	A0	00	:BA
3948	80	80	01	FF	80	07	FA	E0	:61
3950	3E	FA	BC	EE	FA	AF	FF	FB	:85
3958	FF	00	F3	80	FF	81	FF	FF	:F0
3960	80	FA	2F	FE	A8	07	80	A0	:76
3968	00	80	80	01	FF	80	07	FA	:81
3970	E0	3E	FA	BC	EE	FA	AF	FF	:6A
3978	FB	FF	00	F3	80	FF	81	FF	:EC
3980	FF	80	FA	2F	D2	A8	07	80	:A9
3988	A0	00	80	80	01	FF	80	1E	:3E
3990	8A	F8	7A	AA	2E	E8	82	AB	:E9
3998	EA	80	8B	FF	C3	FF	66	CA	:E6
39A0	66	3A	43	8C	1A	C2	A8	0E	:01
39A8	C3	B0	07	43	60	01	FF	80	:9D
39B0	1E	8A	F8	7A	AA	2E	E8	BE	:98
39B8	AB	EA	FE	8B	FF	FF	FF	66	:81
39C0	CA	66	3A	43	8C	1A	C2	A8	:BD
39C8	0E	C3	B0	07	43	60	01	FF	:2B
39D0	80	1E	8A	F8	7A	AA	2E	E8	:5A
39D8	BA	AB	EA	E0	8B	FF	C3	FF	:7B
39E0	66	CA	66	3A	43	8C	1A	C2	:7B
39E8	A8	0E	C3	B0	07	43	60	00	:D3
39F0	00	00	00	00	00	00	00	00	:00
39F8	00	00	00	80	82	81	AA	E7	:14
3A00	AB	8F	81	E1	00	00	00	00	:9C
3A08	00	00	00	00	00	00	00	00	:00
3A10	00	00	00	00	00	00	00	00	:00
3A18	00	00	18	00	80	BE	81	AA	:81
3A20	FF	AB	8F	81	E1	00	00	00	:9B
3A28	00	00	00	00	00	00	00	00	:00
3A30	00	00	00	00	00	00	00	00	:00
3A38	00	00	00	10	00	80	A2	81	:B3
3A40	AA	E7	AB	8F	81	E1	00	00	:2D
3A48	00	00	00	00	00	00	00	00	:00
3A50	00	00	00	00	00	00	00	00	:00
3A58	FF	FF	FF	6A	AA	AA	3A	AA	:9F
3A60	A8	1A	AA	A8	0E	AA	A0	06	:72
3A68	AA	A0	03	AA	80	00	00	00	:77
3A70	00	00	00	00	00	00	00	00	:00
3A78	00	FF	FF	FF	6A	AA	AA	3A	:F5
3A80	AA	A8	1A	AA	A8	0E	AA	A0	:16
3A88	06	AA	A0	03	AA	80	00	00	:7D
3A90	00	00	00	00	00	00	00	00	:00
3A98	00	00	FF	FF	FF	6A	AA	AA	:BB
3AA0	3A	AA	A8	1A	AA	A8	0E	AA	:B0
3AA8	A0	06	AA	A0	03	AA	80	00	:1D
3AB0	00	00	00	00	00	00	00	00	:00
3AB8	00	00	00	03	FF	C0	07	FF	:C8
3AC0	E0	0F	FF	F0	1F	FF	F8	3F	:33
3AC8	FF	FC	7F	FF	FE	AA	AA	AA	:75
3AD0	00	00	00	00	00	00	00	00	:00
3AD8	00	00	00	00	03	FF	C0	07	:C9
3AE0	FF	E0	0F	FF	F0	1F	FF	F8	:F3
3AE8	3F	FF	FC	7F	FF	FE	AA	AA	:0A
3AF0	AA	00	00	00	00	00	00	00	:AA
3AF8	00	00	00	00	00	03	FF	C0	:C2
3B00	07	FF	E0	0F	FF	F0	1F	FF	:02
3B08	F8	3F	FF	FC	7F	FF	FE	AA	:58
3B10	AA	AA	00	00	00	00	00	00	:54
3B18	00	18	00	00	38	00	00	6A	:BA
3B20	00	00	FF	00	00	20	00	00	:1F
3B28	00	00	00	20	00	00	FF	00	:1F
3B30	00	6A	00	00	38	00	00	18	:BA
3B38	00	00	18	00	00	38	00	00	:50
3B40	6A	00	00	FF	00	00	38	00	:A1
3B48	00	3C	00	00	38	00	00	FF	:73
3B50	00	00	6A	00	00	38	00	00	:A2
3B58	18	00	00	18	00	00	38	00	:68
3B60	00	6A	00	00	FF	00	00	38	:A1
3B68	00	00	20	00	00	20	00	00	:40
3B70	FF	00	00	6A	00	00	38	00	:A1
3B78	00	18	00	00	00	00	00	18	:30
3B80	00	00	38	00	00	6A	00	00	:A2
3B88	FF	00	00	00	00	00	FF	00	:FE
3B90	00	6A	00	00	38	00	00	18	:BA
3B98	00	00	00	00	00	00	00	00	:00
3BA0	18	00	00	38	00	00	6A	00	:BA
3BA8	00	FF	00	00	3C	00	00	FF	:3A
3BB0	00	00	6A	00	00	38	00	00	:A2
3BB8	18	00	00	00	00	00	00	00	:18
3BC0	00	18	00	00	38	00	00	6A	:BA
3BC8	00	00	FF	00	00	10	00	00	:0F
3BD0	FF	00	00	6A	00	00	38	00	:A1
3BD8	00	18	00	00	00	00	00	00	:18
3BE0	00	00	00	00	00	00	00	00	:00
3BE8	38	00	00	E2	00	00	C2	00	:DC
3BF0	00	E2	00	00	28	00	00	00	:0A
3BF8	00	00	00	00	00	00	00	00	:00
3C00	00	00	00	00	00	00	00	00	:00
3C08	00	38	00	00	FA	00	00	FE	:30
3C10	00	00	FA	00	00	28	00	00	:22
3C18	00	00	00	00	00	00	00	00	:00
3C20	00	00	00	00	00	00	00	00	:00
3C28	00	00	38	00	00	FA	00	00	:32
3C30	E2	00	00	E2	00	00	28	00	:EC
3C38	00	00	00	00	00	00	00	00	:00
3C40	00	03	FF	C0	1C	10	38	68	:8E
3C48	92	16	82	54	41	90	81	09	:D9
3C50	AA	81	55	90	81	09	82	54	:70
3C58	41	68	92	16	1C	10	38	03	:B8
3C60	FF	C0	03	FF	C0	1C	10	38	:E5
3C68	68	92	16	82	54	41	90	99	:50
3C70	09	AA	BD	55	90	99	09	82	:79
3C78	54	41	68	92	16	1C	10	38	:09
3C80	03	FF	C0	03	FF	C0	1C	10	:B0
3C88	38	68	92	16	82	54	41	90	:EF
3C90	99	09	AA	A1	55	90	81	09	:5C
3C98	82	54	41	68	92	16	1C	10	:53
3CA0	38	03	FF	C0	00	00	00	00	:FA
3CA8	00	00	00	00	00	00	00	00	:00

Add	+0	+1	+2	+3	+4	+5	+6	+7	Sum
4200	07	00	00	00	00	00	00	00	:07
4208	00	00	00	00	00	00	00	00	:00
4210	00	01	00	00	00	00	00	00	:01
4218	00	00	00	00	00	00	00	00	:00
4220	00	00	00	00	00	00	00	00	:00
4228	00	00	00	00	00	00	00	00	:00
4230	00	00	00	00	00	00	00	00	:00
4238	00	00	00	00	00	00	00	00	:00
4240	05	0F	14	19	1E	23	28	2D	:D7
4248	00	00	00	00	00	00	00	00	:00
4250	0A	14	1E	28	0A	14	1E	28	:C8
4258	00	00	00	00	00	00	00	00	:00
4260	00	00	00	00	00	00	00	00	:00
4268	00	00	00	00	00	00	00	00	:00
4270	01	01	01	01	01	01	01	01	:08
4278	00	00	00	00	00	00	00	00	:00
4280	07	00	00	00	00	00	00	00	:07
4288	00	00	00	00	00	00	00	00	:00
4290	00	01	08	00	00	00	00	00	:09
4298	00	00	00	00	00	00	00	00	:00
42A0	00	00	00	00	00	00	00	00	:00
42A8	00	00	00	00	00	00	00	00	:00
42B0	00	00	00	00	00	00	00	00	:00
42B8	00	00	00	00	00	00	00	00	:00
42C0	05	0F	14	19	1E	23	28	2D	:D7
42C8	00	00	00	00	00	00	00	00	:00
42D0	0A	14	1E	28	0A	14	1E	28	:C8
42D8	00	00	00	00	00	00	00	00	:00
42E0	00	00	00	00	00	00	00	00	:00
42E8	00	00	00	00	00	00	00	00	:00
42F0	01	00	01	01	01	00	01	01	:06
42F8	00	00	00	00	00	00	00	00	:00
4300	07	00	00	00	00	00	00	00	:07
4308	00	00	00	00	00	00	00	00	:00
4310	00	01	00	00	01	02	03	04	:0B
4318	00	00	00	00	00	00	00	00	:00
4320	00	00	00	00	00	00	00	00	:00
4328	00	00	00	00	00	00	00	00	:00
4330	00	00	00	00	00	00	00	00	:00
4338	00	00	00	00	00	00	00	00	:00
4340	05	0F	14	19	1E	23	28	2D	:D7
4348	00	00	00	00	00	00	00	00	:00
4350	0A	14	1E	28	0A	14	1E	28	:C8
4358	00	00	00	00	00	00	00	00	:00
4360	00	00	00	00	00	00	00	00	:00
4368	00	00	00	00	00	00	00	00	:00
4370	01	00	01	00	01	00	01	00	:04
4378	00	00	00	00	00	00	00	00	:00
4380	07	00	00	00	00	00	00	00	:07
4388	00	00	00	00	00	00	00	00	:00
4390	03	00	02	03	00	00	00	00	:08
4398	00	00	00	00	00	00	00	00	:00
43A0	00	00	00	00	00	00	00	00	:00
43A8	00	00	00	00	00	00	00	00	:00
43B0	02	02	02	02	02	02	02	02	:10
43B8	00	00	00	00	00	00	00	00	:00
43C0	0A	14	1E	28	0F	19	23	2D	:DC
43C8	00	00	00	00	00	00	00	00	:00
43D0	0A	28	46	64	14	32	50	6E	:E0
43D8	00	00	00	00	00	00	00	00	:00
43E0	03	04	03	04	03	04	03	04	:1C
43E8	00	00	00	00	00	00	00	00	:00
43F0	01	00	01	01	01	01	01	00	:06
43F8	00	00	00	00	00	00	00	00	:00
4400	07	00	00	00	00	00	00	00	:07
4408	00	00	00	00	00	00	00	00	:00
4410	00	01	02	03	04	05	06	07	:1C
4418	00	00	00	00	00	00	00	00	:00
4420	00	00	00	00	00	00	00	00	:00
4428	00	00	00	00	00	00	00	00	:00
4430	00	00	00	00	02	02	02	02	:08
4438	00	00	00	00	00	00	00	00	:00
4440	0A	14	1E	28	0F	19	23	2D	:DC
4448	08	10	18	20	08	10	18	20	:A0
4450	0A	14	1E	28	14	32	50	6E	:68
4458	32	32	32	32	41	41	41	41	:CC
4460	00	00	00	00	03	04	03	04	:0E
4468	04	04	04	04	04	04	04	04	:20
4470	01	00	01	01	01	00	01	01	:06
4478	01	00	01	01	00	00	00	00	:04
4480	07	00	00	00	00	00	00	00	:07
4488	00	00	00	00	00	00	00	00	:00
4490	06	07	00	00	06	07	00	00	:1A
4498	00	00	00	00	00	00	00	00	:00
44A0	01	01	01	01	01	01	01	01	:08
44A8	00	00	00	00	00	00	00	00	:00
44B0	04	04	04	04	04	04	04	04	:20
44B8	00	00	00	00	00	00	00	00	:00
44C0	05	05	05	05	05	05	05	05	:28
44C8	08	10	18	20	08	10	18	20	:A0
44D0	0A	0A	0A	0A	0A	0A	0A	0A	:50
44D8	32	32	32	32	41	41	41	41	:CC
44E0	03	03	03	03	04	04	04	04	:1C
44E8	04	04	04	04	04	04	04	04	:20
44F0	01	01	01	01	01	01	01	01	:08
44F8	01	01	01	01	00	00	00	00	:04
4500	0F	00	00	00	00	00	00	00	:0F
4508	00	00	00	00	00	00	00	00	:00
4510	00	00	00	00	06	07	00	00	:0D
4518	00	00	00	00	00	00	00	00	:00
4520	01	01	01	01	01	01	01	01	:08
4528	01	01	01	01	01	01	01	01	:08
4530	04	04	04	04	04	04	04	04	:20
4538	04	04	04	04	04	04	04	04	:20
4540	05	05	05	05	05	05	05	05	:28
4548	37	37	37	37	37	37	37	37	:B8
4550	0A	0A	0A	0A	0A	0A	0A	0A	:50
4558	0A	0A	0A	0A	0A	0A	0A	0A	:50
4560	03	03	03	03	03	03	03	03	:18
4568	04	04	04	04	04	04	04	04	:20
4570	01	01	01	01	01	01	01	01	:08
4578	01	01	01	01	01	01	01	01	:08
4580	07	00	00	00	00	00	00	00	:07
4588	00	00	00	00	00	00	00	00	:00
4590	00	01	00	00	06	07	00	00	:0E
4598	00	00	00	00	00	00	00	00	:00
45A0	01	01	01	01	01	01	01	01	:08
45A8	00	00	00	00	00	00	00	00	:00
45B0	00	00	00	00	04	04	04	04	:10
45B8	00	00	00	00	00	00	00	00	:00
45C0	05	05	05	05	05	05	05	05	:28
45C8	00	00	00	00	00	00	00	00	:00
45D0	0A	0A	0A	0A	0A	0A	0A	0A	:50
45D8	00	00	00	00	00	00	00	00	:00
45E0	00	00	00	00	03	04	03	04	:0E
45E8	00	00	00	00	00	00	00	00	:00
45F0	01	00	01	00	01	01	01	01	:06
45F8	00	00	00	00	00	00	00	00	:00
4600	07	00	00	00	00	00	00	00	:07
4608	00	00	00	00	00	00	00	00	:00
4610	00	00	02	03	06	07	00	00	:12
4618	00	00	00	00	00	00	00	00	:00
4620	01	01	01	01	01	01	01	01	:08
4628	00	00	00	00	00	00	00	00	:00
4630	02	02	02	02	04	04	04	04	:18
4638	00	00	00	00	00	00	00	00	:00
4640	05	05	05	05	05	05	05	05	:28
4648	00	00	00	00	00	00	00	00	:00
4650	0A	0A	0A	0A	0A	0A	0A	0A	:50
4658	00	00	00	00	00	00	00	00	:00
4660	03	04	03	04	03	04	03	04	:1C
4668	00	00	00	00	00	00	00	00	:00
4670	01	00	01	00	01	01	01	01	:06

```
4678  00 00 00 00 00 00 00 00 :00
4680  07 00 00 00 00 00 00 00 :07
4688  00 00 00 00 00 00 00 00 :00
4690  00 00 00 00 00 00 04 05 :09
4698  00 00 00 00 00 00 00 00 :00
46A0  00 00 00 00 00 00 00 00 :00
46A8  00 00 00 00 00 00 00 00 :00
46B0  06 06 06 06 06 06 06 06 :30
46B8  00 00 00 00 00 00 00 00 :00
46C0  1E 14 14 1E 28 32 32 28 :18
46C8  00 00 00 00 00 00 00 00 :00
46D0  46 32 1E 0A 0A 1E 32 46 :40
46D8  00 00 00 00 00 00 00 00 :00
46E0  05 05 05 05 05 05 05 05 :28
46E8  00 00 00 00 00 00 00 00 :00
46F0  00 00 00 00 00 00 00 00 :00
46F8  00 00 00 00 00 00 00 00 :00
4700  07 00 00 00 00 00 00 00 :07
4708  00 00 00 00 00 00 00 00 :00
4710  00 00 00 00 00 00 0A 0B :15
4718  00 00 00 00 00 00 00 00 :00
4720  00 00 00 00 00 00 00 00 :00
4728  00 00 00 00 00 00 00 00 :00
4730  06 06 06 06 06 06 06 06 :30
4738  00 00 00 00 00 00 00 00 :00
4740  1E 14 14 1E 28 32 32 28 :18
4748  00 00 00 00 00 00 00 00 :00
4750  46 32 1E 0A 0A 1E 32 46 :40
4758  00 00 00 00 00 00 00 00 :00
4760  01 02 01 02 01 02 01 02 :0C
4768  00 00 00 00 00 00 00 00 :00
4770  01 00 01 01 01 01 00 01 :06
4778  00 00 00 00 00 00 00 00 :00
4780  07 00 00 00 00 00 00 00 :07
4788  00 00 00 00 00 00 00 00 :00
4790  00 00 00 00 00 00 0A 0B :15
4798  00 00 00 00 00 00 00 00 :00
47A0  00 00 00 00 00 00 00 00 :00
47A8  00 00 00 00 00 00 00 00 :00
47B0  06 06 06 06 06 06 06 06 :30
47B8  00 00 00 00 00 00 00 00 :00
47C0  0A 14 1E 28 0F 19 23 2D :DC
47C8  00 00 00 00 00 00 00 00 :00
47D0  0A 28 46 64 14 32 50 6E :E0
47D8  00 00 00 00 00 00 00 00 :00
47E0  02 02 02 02 02 02 02 02 :10
47E8  00 00 00 00 00 00 00 00 :00
47F0  01 00 01 01 01 01 00 01 :06
47F8  00 00 00 00 00 00 00 00 :00
4800  0F 00 00 00 00 00 00 00 :0F
4808  00 00 00 00 00 00 00 00 :00
4810  00 00 00 00 08 09 00 00 :11
4818  00 00 00 00 00 00 00 00 :00
4820  01 01 01 01 01 01 01 01 :08
4828  01 01 01 01 01 01 01 01 :08
4830  04 04 04 04 04 04 04 04 :20
4838  04 04 04 04 04 04 04 04 :20
4840  05 05 05 05 05 05 05 05 :28
4848  37 37 37 37 37 37 37 37 :B8
4850  0A 0A 0A 0A 0A 0A 0A 0A :50
4858  0A 0A 0A 0A 0A 0A 0A 0A :50
4860  03 03 03 03 03 03 03 03 :18
4868  04 04 04 04 04 04 04 04 :20
4870  00 00 00 00 00 00 00 00 :00
4878  00 00 00 00 00 00 00 00 :00
4880  0F 00 00 00 00 00 00 00 :0F
4888  00 00 00 00 00 00 00 00 :00
4890  00 01 02 03 04 05 0A 0B :24
4898  00 00 00 00 00 00 00 00 :00
48A0  00 00 00 00 00 00 00 00 :00
48A8  00 00 00 00 00 00 00 00 :00
48B0  00 00 00 00 02 02 02 02 :08
48B8  04 04 04 04 06 06 06 06 :28
48C0  05 05 05 05 05 05 05 05 :28
48C8  37 37 37 37 37 37 37 37 :B8
48D0  0A 0A 0A 0A 0A 0A 0A 0A :50
48D8  0A 0A 0A 0A 0A 0A 0A 0A :50
48E0  00 00 00 00 03 04 03 04 :0E
48E8  05 05 05 05 01 02 01 02 :1A
48F0  01 00 01 01 01 00 01 01 :06
48F8  00 00 00 00 01 00 01 00 :02

Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
4A00  34 08 86 53 1F 8B 7F 53 :91
4A08  1A 7E 4A 42 12 12 12 12 :6C
4A10  12 12 12 12 12 12 12 12 :90
4A18  12 12 12 12 12 12 12 12 :90
4A20  12 12 12 12 12 12 12 12 :90
4A28  12 12 12 12 12 12 12 12 :90
4A30  12 12 12 12 12 12 12 12 :90
4A38  12 12 12 12 12 12 12 12 :90
4A40  12 12 8E 52 80 31 88 17 :54
4A48  BD 50 D4 8E 52 98 31 88 :12
4A50  17 BD 50 D4 0F 04 CE 00 :D9
4A58  33 EE C4 11 B3 00 35 10 :EE
4A60  24 04 E1 EC 42 10 83 00 :CA
4A68  64 26 EE 11 B3 00 35 10 :81
4A70  24 04 8C DF 00 EC C4 DD :20
4A78  02 33 43 33 41 31 41 10 :6E
4A80  9C 02 25 04 1F 23 20 E3 :0C
4A88  A6 C4 81 3A 27 ED 8E 52 :19
4A90  C0 10 8E 00 00 5F A6 C5 :28
4A98  A1 85 27 0D 30 03 31 22 :E0
4AA0  8C 52 E2 10 22 02 FE 20 :12
4AA8  EC 5C C1 03 2D E8 31 A9 :FB
4AB0  53 A4 33 43 EC 84 DD 10 :CA
4AB8  A6 02 97 12 6E B4 BD 4D :7D
4AC0  57 86 17 BD 4D 71 97 13 :19
4AC8  8E 52 80 E6 86 10 26 03 :05
4AD0  3B 6C 86 C6 05 3D C3 52 :4A
4AD8  00 1F 02 BD 4D 5B BD 4D :90
4AE0  67 BD 4D 5F 86 04 BD 4D :64
4AE8  71 A7 A4 BD 4D 6B 86 04 :BB
4AF0  BD 4D 71 81 00 27 1A 81 :BE
4AF8  01 26 04 86 80 20 12 81 :E4
4B00  02 26 04 86 C0 20 0A 81 :1D
4B08  03 26 04 86 E0 20 02 86 :3B
4B10  F0 A7 21 BD 4D 6B 86 04 :B7
4B18  BD 4D 71 81 00 27 06 C6 :EF
4B20  02 3D 5A 1F 98 A7 22 BD :D6
4B28  4D 6B 86 01 BD 4D 71 A7 :61
4B30  23 BD 4D 6B 86 01 BD 4D :29
4B38  71 A7 24 BD 4D 63 7E 4A :71
4B40  7D 96 04 10 26 03 44 0C :A0
4B48  04 8E 52 B8 10 8E 52 BD :49
4B50  BD 50 D4 8E 52 A8 10 8E :07
4B58  52 B7 BD 50 D4 BD 4D 57 :4B
4B60  86 0F BD 4D 71 97 13 8E :48
4B68  52 98 E6 86 10 26 02 85 :13
4B70  97 05 5F 44 56 C3 42 00 :9A
4B78  DD 06 BD 4D 5B 7E 4A 7D :8D
4B80  BD 4D 8B 4F BD 4D 95 BD :40
4B88  4D 67 86 10 BD 4D 71 4A :0F
4B90  A7 9F 53 06 97 09 7E 4A :07
4B98  7D BD 4D 8B BD 4D 57 86 :F9
4BA0  07 BD 4D 71 97 13 8E 52 :0C
4BA8  A8 E6 86 10 26 02 74 6C :2C
4BB0  86 97 08 BD 4D 5B BD 4D :94
4BB8  67 86 17 BD 4D 71 10 9E :2D
4BC0  06 31 A8 10 D6 08 A7 A5 :19
4BC8  7E 4A 7D BD 4D 8B 86 01 :61
4BD0  BD 4D 95 10 9E 06 31 A8 :2C
4BD8  20 BD 4D 5F 0F 0A 86 01 :29
4BE0  BD 4D 71 D6 0A A7 A5 96 :3D
4BE8  0A 91 09 24 05 BD 4D 6B :42
4BF0  20 EC BD 4D 63 7E 4A 7D :BE
4BF8  BD 4D 8B 86 02 BD 4D 95 :BC
4C00  BD 4D 5F 10 9E 06 31 A8 :F6
4C08  30 86 07 97 0B 8E 52 A8 :E7
4C10  7E 4C 2B BD 4D 8B 86 04 :14
4C18  BD 4D 95 BD 4D 5F 10 9E :B6
4C20  06 31 A8 60 86 17 97 0B :7E
4C28  8E 52 80 0F 0A 96 0B BD :D7
4C30  4D 71 97 13 E6 86 26 0B :05
4C38  96 0B 81 07 10 26 02 17 :78
4C40  16 02 2B D6 0A A7 A5 96 :05
4C48  0A 91 09 24 05 BD 4D 6B :42
4C50  20 DB BD 4D 63 7E 4A 7D :AD
4C58  BD 4D 8B 86 03 BD 4D 95 :BD
4C60  10 9E 06 31 A8 40 0F 0A :E6
4C68  BD 4D 57 86 39 BD 4D 71 :9B
4C70  D6 0A A7 A5 A6 C0 81 2C :3F
4C78  10 26 01 37 86 B0 BD 4D :AE
4C80  71 D6 0A CB 10 A7 A5 BD :35
4C88  4D 5B 96 0A 91 09 24 05 :0B
4C90  BD 4D 6B 20 D3 7E 4A 7D :AD
4C98  BD 4D 8B 86 05 BD 4D 95 :BF
4CA0  10 9E 06 6F 21 31 A8 70 :8D
4CA8  7E 4B D9 BD 4D 8B 86 05 :C2
4CB0  BD 4D 95 10 9E 06 86 01 :DA
4CB8  A7 21 7E 4A 7D 96 04 10 :B7
4CC0  27 01 F0 0F 04 8E 52 B8 :C3
4CC8  8C 52 BD 22 08 A6 80 10 :FB
4CD0  27 02 1B 20 F3 8E 52 98 :CF
4CD8  D6 05 86 01 A7 85 7E 4A :56
4CE0  7D 96 04 10 26 01 A4 A6 :98
4CE8  C0 81 3A 10 26 00 C4 8E :03
4CF0  53 1E 10 8E 52 98 0F 0D :15
4CF8  96 0D 81 40 10 24 02 19 :B3
4D00  A6 C4 81 44 26 0A 86 DD :C2
4D08  D6 0D A7 85 33 41 20 11 :B4
4D10  86 0F BD 4D 71 97 13 E6 :A0
4D18  A6 10 27 01 23 D6 0D A7 :8B
4D20  85 0C 0D A6 C0 81 DA 27 :86
4D28  CF 81 3A 10 26 00 84 EC :30
4D30  C1 10 83 46 49 10 26 01 :1A
4D38  EE A6 C4 81 4E 10 26 01 :5E
4D40  E6 D6 0D 86 FF A7 85 35 :AF
4D48  88 97 0C A6 C0 91 0C 27 :55
4D50  05 CC 4D B3 ED E4 39 86 :61
4D58  28 20 EE 86 29 20 EA 86 :75
4D60  5B 20 E6 86 5D 20 E2 86 :CC
4D68  E6 20 DE 0C 0A 86 2C 20 :CC
4D70  D8 97 0C A6 C0 81 FE 26 :86
4D78  D8 A6 C0 81 01 26 06 A6 :92
4D80  C0 91 0C 23 05 CC 4D C7 :65
4D88  ED E4 39 96 04 26 05 CC :9B
4D90  4E B3 ED E4 39 8E 52 B8 :A3
4D98  E6 86 26 03 6C 86 39 CC :8C
4DA0  4D DB ED E4 39 8E 51 A0 :B1
4DA8  C6 08 BD 50 A2 BD 4F 74 :FD
4DB0  7E 4F 4C BD 50 0D BD 50 :40
4DB8  20 8E 51 A0 C6 08 BD 50 :7A
4DC0  A2 BD 4F 74 7E 4F 4C BD :F8
4DC8  50 0D BD 50 20 8E 51 A8 :11
4DD0  C6 09 BD 50 A2 BD 4F 74 :FE
4DD8  7E 4F 4C BD 4F 54 BD 50 :86
4DE0  20 BD 50 0D BD 50 31 8E :06
4DE8  51 B1 C6 0A BD 50 A2 BD :3E
4DF0  4F 74 7E 4F 4C 8E 51 B1 :6C
4DF8  C6 0A BD 50 A2 BD 50 07 :93
4E00  BD 50 3D BD 50 7C BD 4F :DF
4E08  74 7E 4F 4C 8E 51 B1 C6 :E3
4E10  0A BD 50 A2 BD 50 07 BD :8A
4E18  50 52 BD 50 7C BD 4F 74 :AB
4E20  7E 4F 4C BD 4F 54 BD 50 :86
4E28  20 BD 50 67 BD 50 7C BD :DA
4E30  50 31 8E 51 B1 C6 0A BD :9E
4E38  50 A2 BD 4F 74 7E 4F 4C :8B
4E40  8E 51 BB C6 0C BD 50 A2 :1B
4E48  BD 50 07 BD 50 3D BD 50 :6B
4E50  7C BD 4F 74 7E 4F 4C 8E :A3
4E58  51 BB C6 0C BD 50 A2 BD :4A
4E60  50 07 BD 50 52 BD 50 7C :3F
4E68  BD 4F 74 7E 4F 4C BD 4F :A5
4E70  54 BD 50 20 BD 50 67 BD :B2
4E78  50 7C BD 50 31 8E 51 BB :A4
4E80  C6 0C BD 50 A2 BD 4F 74 :01
4E88  7E 4F 4C BD 4F 54 BD 50 :86
4E90  07 86 C6 BD 50 07 86 45 :32
4E98  BD 50 8B 86 4E BD 50 8B :04
4EA0  86 50 BD 50 8B BD 50 07 :82
4EA8  8E 51 C7 C6 04 BD 50 A2 :1F
4EB0  7E 4F 4C 86 50 BD 50 8B :87
4EB8  86 41 BD 50 8B C6 0E BD :F0
4EC0  50 A2 7E 4F 4C 8E 51 C7 :B1
4EC8  C6 04 BD 50 A2 86 C9 BD :85
4ED0  50 8B 86 C6 BD 50 8B BD :7C
4ED8  50 07 BD 50 0D 8E 51 CB :1B
4EE0  8E 51 CB C6 04 BD 50 A2 :23
4EE8  BD 4F 74 7E 4F 4C BD 4F :A5
4EF0  54 BD 50 07 8E 51 CF C6 :DC
4EF8  0E BD 50 A2 7E 4F 4C 86 :5C
4F00  42 BD 50 8B 86 47 BD 50 :B4
4F08  8B 86 4E BD 50 8B 8E 51 :D6
4F10  C7 C6 04 BD 50 A2 7E 4F :0D
4F18  4C 8E 51 DD C6 0D BD 50 :E8
4F20  A2 BD 4F 74 7E 4F 4C 86 :C1
4F28  46 BD 50 8B 86 49 BD 50 :BA
4F30  8B 86 4E BD 50 8B 8E 51 :D6
4F38  C7 C6 04 BD 50 A2 BD 4F :4C
4F40  74 7E 4F 4C 8E 51 90 C6 :C2
4F48  10 BD 50 A2 86 07 BD 50 :59
4F50  8B 7E 50 EA 86 50 BD 50 :26
4F58  8B 86 41 BD 50 8B 86 54 :C4
4F60  BD 50 8B 86 28 BD 50 8B :DE
4F68  D6 05 1D BD 4F 8D 86 29 :40
4F70  BD 50 8B 39 BD 50 07 86 :6B
4F78  49 BD 50 8B 86 6E BD 50 :E2
4F80  8B BD 50 07 10 9E 00 EC :39
4F88  22 BD 4F 8D 39 10 83 00 :87
4F90  00 26 06 86 30 BD 50 8B :7A
4F98  39 34 10 8E 53 90 ED 09 :E4
```

```
4FA0  6F 06 6F 05 A6 06 26 05 :C0
4FA8  CC 27 10 20 22 81 01 26 :ED
4FB0  05 CC 03 E8 20 19 81 02 :78
4FB8  26 05 CC 00 64 20 10 81 :0C
4FC0  03 26 05 CC 00 0A 20 07 :2B
4FC8  81 04 26 1E CC 00 01 ED :83
4FD0  07 EC 09 10 A3 07 25 08 :E3
4FD8  A3 07 ED 09 6C 05 20 F3 :24
4FE0  A6 05 E6 06 A7 85 6C 06 :35
4FE8  20 B8 5F 0F 0C A6 85 27 :A4
4FF0  04 0C 0C 20 06 0D 0C 26 :81
4FF8  02 86 D0 8B 30 BD 50 8B :AB
5000  5C C1 04 2F E8 35 90 86 :83
5008  20 BD 50 8B 39 8E 53 10 :E2
5010  A6 80 BD 50 8B A6 80 BD :A1
5018  50 8B A6 80 BD 50 8B 39 :D2
5020  BD 50 07 86 C3 BD 50 8B :F5
5028  86 DE BD 50 8B BD 50 07 :10
5030  39 BD 50 07 86 CA BD 50 :AA
5038  8B BD 50 07 39 86 50 BD :6B
5040  50 8B 86 41 BD 50 8B 86 :C0
5048  54 BD 50 8B 86 28 BD 50 :A7
5050  8B 39 86 4D BD 50 8B 86 :B5
5058  56 BD 50 8B 86 50 BD 50 :D1
5060  8B 86 28 BD 50 8B 39 86 :90
5068  43 BD 50 8B 86 48 BD 50 :B6
5070  8B 86 52 BD 50 8B 86 28 :A9
5078  BD 50 8B 39 D6 13 1D BD :94
5080  4F 8D 86 29 BD 50 8B BD :E0
5088  50 07 39 34 16 BD 6F 47 :4D
5090  8E FC 82 C6 03 E7 80 C6 :02
5098  01 E7 80 A7 84 7F FD 05 :14
50A0  35 96 34 56 BD 6F 47 CE :96
50A8  FC 82 86 03 A7 C0 E7 C0 :15
50B0  4F B1 FC 83 22 0D E6 86 :1A
50B8  C1 20 24 02 C6 20 E7 C6 :9A
50C0  4C 20 EE 7F FD 05 35 D6 :E6
50C8  34 72 A6 80 A7 C0 AC 63 :42
50D0  23 F8 35 F2 34 30 6F 80 :95
50D8  AC 62 23 FA 35 B0 34 32 :76
50E0  86 01 A7 80 AC 63 23 F8 :D8
50E8  35 B2 86 01 97 1A 7E 4D :EA
50F0  47 00 00 00 00 00 00 00 :47
50F8  00 00 00 00 00 00 00 00 :00
5100  0C 4F 68 21 46 4D 20 53 :EA
5108  41 4D 50 4C 45 41 52 53 :55
5110  43 20 20 20 20 05 4F 68 :7F
5118  21 46 4D 53 48 4F 50 69 :57
5120  20 20 20 20 20 1E 5B 34 :4D
5128  5D 20 1D 20 EC 20 1C 20 :02
5130  5B 36 5D 20 20 20 42 45 :D5
5138  41 4D A5 A5 A5 5B 42 52 :6C
5140  45 41 4B 5D 0B 48 69 74 :5E
5148  20 5B 53 5D 20 6B 65 79 :94
5150  13 4F 52 49 47 49 4E 41 :1C
5158  4C 20 53 50 41 43 45 20 :F8
5160  47 41 4D 45 0A 41 2E 44 :D7
5168  2E 41 20 73 6F 66 74 03 :4E
5170  42 79 20 05 53 43 4F 52 :17
5178  45 08 48 49 2D 53 43 4F :F0
5180  52 45 09 47 41 4D 45 20 :DA
5188  4F 56 45 52 00 00 00 00 :3C
5190  CC DF DB B8 DE D7 D1 CA :8E
5198  31 30 30 B6 D7 C3 DE BD :7C
51A0  CC DE DD CE DF B3 D0 BD :74
51A8  BD B3 C1 B6 DE B5 B6 BC :EC
51B0  B2 BD C3 DE C6 C3 B2 B7 :02
51B8  DE BC C0 CF C0 DE C3 B2 :3C
51C0  B7 DE BC C3 B2 C5 B2 B6 :F3
51C8  DE C5 B2 B6 DE B1 D9 C9 :3C
51D0  20 C3 B2 B7 DE B6 DE CC :8A
51D8  B6 DD BE DE DD CC DF DB :92
51E0  BE BD B6 DE B5 B5 BD B7 :ED
51E8  DE D9 00 00 00 00 00 00 :B7
51F0  00 00 00 00 00 00 00 00 :00
51F8  00 00 00 00 00 00 00 00 :00
5200  03 00 00 01 01 01 80 00 :86
5208  01 01 02 80 00 01 01 00 :86
5210  80 00 01 01 00 C0 00 01 :43
5218  01 04 80 00 00 01 00 00 :86
5220  00 00 00 00 00 00 00 00 :00
5228  00 00 00 00 00 00 00 00 :00
5230  00 00 00 00 00 00 00 00 :00
5238  00 00 00 00 00 00 00 00 :00
5240  00 00 00 00 00 00 00 00 :00
5248  00 00 00 00 00 00 00 00 :00
5250  00 00 00 00 00 00 00 00 :00
5258  00 00 00 00 00 00 00 00 :00
```

```
5260  00 00 00 00 00 00 00 00 :00
5268  00 00 00 00 00 00 00 00 :00
5270  00 00 00 00 00 00 00 00 :00
5278  00 00 00 00 00 00 00 00 :00
5280  01 01 01 01 01 01 00 00 :06
5288  00 00 00 00 00 00 00 00 :00
5290  00 00 00 00 00 00 00 00 :00
5298  01 01 01 01 01 01 01 01 :08
52A0  01 01 01 01 01 01 00 00 :06
52A8  01 01 01 01 01 01 01 01 :08
52B0  00 00 00 00 00 00 00 00 :00
52B8  01 01 01 01 01 01 00 00 :06
52C0  4D 56 50 50 41 54 43 48 :63
52C8  52 4E 55 4D 44 49 46 43 :58
52D0  48 49 49 4E 50 4D 4F 56 :6A
52D8  54 41 4D 44 41 4D 45 4E :47
52E0  50 42 47 4E 00 00 00 00 :27
52E8  00 00 00 00 00 00 00 00 :00
52F0  00 00 00 00 00 00 00 00 :00
52F8  00 00 00 00 00 00 00 00 :00
5300  20 62 20 B2 00 0D 48 80 :29
5308  07 0F 0F 17 0F 13 00 00 :5E
5310  42 47 4E 0D 00 00 01 E0 :C5
5318  00 00 00 00 00 00 00 03 :03
5320  01 DD 05 04 06 DD 06 07 :D7
5328  08 DD 09 02 0A DD 0B 0C :EE
5330  0D FF 00 00 00 00 00 00 :0C
5338  00 00 00 00 00 00 00 00 :00
5340  00 00 00 00 00 00 00 00 :00
5348  00 00 00 00 00 00 00 00 :00
5350  00 00 00 00 00 00 00 00 :00
5358  00 00 00 00 00 00 00 00 :00
5360  00 00 00 00 00 00 00 00 :00
5368  00 00 00 00 00 00 00 00 :00
5370  52 0A 52 0A 52 0A 52 0A :70
5378  52 00 52 05 52 14 52 14 :75
5380  52 14 52 14 52 14 52 14 :98
5388  52 14 52 14 52 14 52 14 :98
5390  00 00 01 06 00 00 05 00 :0C
5398  01 00 00 00 00 00 00 00 :01
53A0  37 20 00 00 4A BE 4B 41 :EB
53A8  4B 99 4B 80 4B CB 4B F8 :08
53B0  4C 58 4C 13 4C 98 4C AB :DE
53B8  4C BD 4C E1 00 00 00 00 :36
53C0  00 01 01 00 00 00 00 00 :02
53C8  00 00 00 00 00 00 00 00 :00
53D0  00 00 00 01 00 00 00 01 :02
53D8  01 01 01 01 01 01 01 01 :08
53E0  01 58 CB 58 DE 58 F0 59 :FB
53E8  02 59 13 00 00 00 00 00 :6E
53F0  01 2C 00 B9 02 76 00 3C :9A
53F8  07 00 02 02 01 00 00 00 :0C
5400  00 00 3F 59 41 4D 41 55 :BC
5408  43 48 49 93 C3 00 90 0A :C4
5410  00 10 B0 00 00 00 10 B0 :80
5418  00 00 00 00 43 80 00 00 :C3
5420  42 10 02 00 04 05 08 00 :65
5428  00 19 00 00 00 01 B1 14 :DF
5430  0A 1E 21 1E 21 0E 27 00 :BD
5438  00 00 00 00 00 00 00 00 :00
5440  00 7C 7C 43 87 6A 6A 00 :96
5448  00 00 00 00 00 00 00 00 :00
5450  00 1E 21 1E 21 0E 27 00 :B3
5458  00 00 00 00 00 00 00 00 :00
5460  00 7C 7C 43 87 6A 6A 00 :96
5468  00 00 00 00 00 00 00 00 :00
5470  00 05 00 00 00 00 00 00 :05
5478  00 64 00 00 00 07 09 00 :74
5480  00 00 00 01 00 00 00 01 :02
5488  00 00 00 00 01 00 00 00 :01
5490  01 57 76 0F 00 00 00 07 :E4
5498  34 FC 19 00 19 00 00 00 :62
54A0  10 B0 10 B0 0D 8D 0D 8D :B4
54A8  2A 09 2A 09 01 E9 01 01 :52
54B0  00 0F 1D 0F 0F 30 AE AE :D6
54B8  C4 C2 30 0F 1D 0F 0F 30 :30
54C0  A8 AE BE BC 30 01 01 01 :03
54C8  01 01 00 36 CC 26 01 06 :31
54D0  00 00 00 00 00 01 02 00 :03
54D8  00 00 00 00 00 00 00 00 :00
54E0  00 00 00 00 00 01 02 00 :03
54E8  00 00 00 00 00 00 00 00 :00
54F0  00 00 78 5A 00 5F 2F 20 :80
54F8  09 03 54 33 54 32 03 5F :7B
5500  00 00 00 00 00 00 02 06 :08
5508  07 02 06 07 02 06 07 02 :27
5510  06 00 00 00 00 00 00 00 :06
5518  00 00 00 00 00 02 06 07 :0F
```

```
5520  02 06 07 00 06 07 02 06 :24
5528  07 02 00 00 00 00 00 00 :09
5530  00 00 00 00 00 06 07 02 :0F
5538  06 07 00 00 00 02 06 07 :1C
5540  02 07 00 00 00 00 00 00 :09
5548  00 00 00 00 06 07 02 06 :15
5550  00 00 00 00 00 00 00 02 :02
5558  06 07 02 00 00 00 00 00 :0F
5560  00 00 00 00 07 02 06 07 :16
5568  00 00 00 00 00 00 00 06 :06
5570  07 02 06 00 00 00 00 00 :0F
5578  00 00 00 00 00 06 07 02 :0F
5580  00 00 00 00 00 00 00 07 :07
5588  02 06 00 00 00 00 00 00 :08
5590  00 00 00 00 00 00 02 06 :08
5598  07 00 00 00 00 00 07 02 :10
55A0  06 00 00 00 00 00 00 00 :06
55A8  00 00 00 00 00 00 00 07 :07
55B0  02 00 00 00 00 00 02 07 :0B
55B8  00 00 00 00 00 00 00 00 :00
55C0  00 00 00 00 00 00 00 00 :00
55C8  06 07 00 00 00 02 07 00 :16
55D0  00 00 00 00 00 00 00 00 :00
55D8  00 00 00 00 00 00 00 00 :00
55E0  00 01 00 00 00 01 00 00 :02
55E8  00 00 00 00 00 00 00 00 :00
55F0  00 00 00 00 00 00 00 00 :00
55F8  01 01 00 00 00 01 01 00 :04
5600  00 00 00 00 00 00 00 00 :00
5608  3A B5 3A D6 3A F7 3A D6 :40
5610  3A F7 3B 18 04 56 51 FD :2C
5618  56 15 8E 56 08 EC 84 ED :B4
5620  06 EC 02 ED 08 EC 04 ED :C6
5628  0A 8E 55 00 8C 56 08 24 :FB
5630  1F CE 56 0E 86 01 AD 9F :24
5638  56 15 CE 56 10 86 02 AD :D4
5640  9F 56 15 CE 56 12 86 04 :CA
5648  AD 9F 56 15 30 08 20 DC :EB
5650  39 B7 56 14 1F 12 C6 08 :59
5658  34 04 4F E6 A0 F5 56 14 :6C
5660  27 04 1A 01 20 02 1C FE :82
5668  49 6A E4 26 EE A7 D4 EC :12
5670  C4 C3 00 01 ED C4 35 84 :F2
5678  B7 56 14 1F 12 C6 08 34 :54
5680  04 A6 D4 34 02 4F E6 E4 :CD
5688  58 89 00 E7 E4 F6 56 14 :0C
5690  3D EA A4 E7 A0 6A 61 26 :43
5698  EC EC C4 C3 00 01 ED C4 :11
56A0  35 86 CC 56 51 16 FF 6F :B2
56A8  CC 56 78 16 FF 69 87 A8 :47
56B0  B6 53 1A 26 0D 34 09 BD :50
56B8  56 D3 35 09 CC 01 E0 FD :11
56C0  FF F6 7E 86 84 12 12 12 :B3
56C8  12 12 12 12 12 12 10 FE :7A
56D0  53 A0 39 10 FF 53 A0 1A :48
56D8  40 CC 6E DC FD FF F6 1C :64
56E0  BF 8E 60 00 10 8E 6C 00 :B7
56E8  CE C0 00 BD 6F 00 BD 6C :E3
56F0  00 8E 54 7A 10 8E 54 7F :CD
56F8  BD 50 D4 0F 22 34 08 86 :D4
5700  54 1F 8B 8E 54 72 10 8E :F0
5708  54 77 BD 50 D4 BD 6C 90 :65
5710  7E 6E C0 12 12 12 12 12 :06
5718  12 12 12 12 12 0F 13 BD :39
5720  6E 0B 86 04 97 9F C6 AC :AB
5728  D7 12 C6 3E C1 4E 24 0B :2B
5730  D7 11 86 0F BD 6F 6C CB :E0
5738  04 20 F1 CE 53 1E A6 C0 :BA
5740  81 FF 27 F7 81 DD 10 27 :33
5748  07 3F 5F 44 56 C3 42 00 :44
5750  DD 1C A6 9F 54 1C 97 24 :69
5758  8E 53 D1 10 8E 53 E0 BD :40
5760  50 DE 31 86 BD 50 D4 8E :54
5768  53 70 10 9E 1C 31 A8 60 :C6
5770  5F 34 04 E6 E4 D1 24 22 :78
5778  0E A6 A0 C6 05 3D C3 52 :71
5780  00 ED 81 6C E4 20 EC 35 :FF
5788  04 34 40 10 9E 1C 31 A8 :1B
5790  10 CE C5 00 0F 26 96 26 :94
5798  81 08 24 2D 8E 38 00 A6 :46
57A0  A0 C6 63 3D 30 8B 0C 26 :F3
57A8  34 60 31 88 20 5F BD 6F :F8
57B0  00 C1 02 27 0D 30 88 21 :D0
57B8  31 A8 21 33 C9 01 08 5C :5B
57C0  20 EC 35 60 33 C8 21 20 :DD
57C8  CD 10 9E 1C 30 A8 20 96 :25
57D0  24 31 86 CE 54 81 BD 50 :8B
57D8  C8 10 9E 1C 30 A8 30 31 :CB
```

```
57E0  0F CE 54 D1 BD 50 C8 33 :0A
57E8  C8 10 BD 50 C8 10 9E 1C :77
57F0  30 A8 40 31 0F CE 54 31 :AB
57F8  BD 50 C8 33 C8 20 BD 50 :FD
5800  C8 30 88 10 31 A8 10 33 :AC
5808  50 BD 50 C8 33 C8 20 BD :FD
5810  50 C8 10 9E 1C 0F 2C 30 :4D
5818  A8 70 31 0F CE 53 C1 BD :F7
5820  50 C8 BD 6B 00 8E 54 C5 :E7
5828  10 8E 54 C9 BD 6E B8 0F :AD
5830  2B CC 00 00 DD 1A 0F 29 :26
5838  8E 00 10 A6 89 53 D0 10 :00
5840  26 01 94 0C 93 1F 10 33 :BC
5848  8B 33 C9 53 6E EE C4 A6 :A0
5850  41 27 07 BD 5D F4 A1 41 :5F
5858  25 49 A6 89 54 80 27 4B :E3
5860  A6 43 26 17 1F 10 C3 00 :18
5868  3F D3 1C 1F 02 A6 A4 A7 :40
5870  89 54 30 86 02 A7 89 54 :19
5878  40 20 11 86 34 BD 5E 1D :63
5880  8B 05 A7 89 54 30 86 02 :CC
5888  A7 89 54 40 A6 42 27 0F :E2
5890  86 07 A0 42 4C BD 5E 1D :F3
5898  AB 42 4A A7 89 54 D0 6F :FA
58A0  89 54 80 A6 89 54 80 10 :70
58A8  26 01 2C A6 44 27 11 A6 :1B
58B0  89 54 D0 44 25 06 6C 89 :11
58B8  54 D0 20 04 6A 89 54 D0 :5F
58C0  A6 C4 48 10 8E 53 E1 31 :B5
58C8  A6 6E B4 A6 89 54 40 8B :16
58D0  04 81 B0 10 24 00 DB A7 :EB
58D8  89 54 40 16 00 90 A6 89 :F2
58E0  54 30 8B 01 81 39 10 24 :FE
58E8  00 B8 A7 89 54 30 20 DB :67
58F0  A6 89 54 30 80 01 81 02 :B7
58F8  10 23 00 AE A7 89 54 30 :95
5900  20 C9 A6 89 54 30 BD 6F :C8
5908  A4 9B 2D 81 39 10 24 00 :5A
5910  91 20 E3 DC 1A 10 83 80 :9D
5918  00 24 E7 C3 00 20 DD 1A :E5
5920  96 29 81 1D 27 2E 81 0E :41
5928  27 2A 24 14 6C 89 54 30 :02
5930  A6 89 54 30 81 39 25 36 :C8
5938  86 02 A7 89 54 30 20 2E :8A
5940  6A 89 54 30 A6 89 54 30 :2A
5948  81 02 22 22 86 39 A7 89 :B6
5950  54 30 20 1A A6 89 54 40 :81
5958  8B 06 A7 89 54 40 A6 89 :84
5960  54 40 81 B0 24 02 20 06 :11
5968  86 02 A7 89 54 40 A6 89 :7B
5970  54 30 E6 89 54 40 DD 16 :7A
5978  A6 89 54 50 E6 89 54 60 :F6
5980  DD 11 A6 89 54 E0 E6 89 :C0
5988  54 D0 DD 14 4F BD 6F 6C :FC
5990  DC 16 A7 89 54 50 E7 89 :36
5998  54 60 96 15 A7 89 54 E0 :C3
59A0  20 29 86 02 A7 89 54 30 :85
59A8  20 C4 86 39 A7 89 54 30 :57
59B0  20 BC A6 89 54 50 E6 89 :1E
59B8  54 60 DD 11 A6 89 54 E0 :05
59C0  97 14 86 0E BD 6F 6C 6C :43
59C8  89 54 80 BD 5A 22 BD 5A :AD
59D0  61 BD 5B 29 BD 5B 7B 30 :65
59D8  1F 10 26 FE 5E 0C 29 96 :7C
59E0  29 81 1D 23 02 0F 29 86 :AA
59E8  07 BD 6F 6C 96 24 10 2C :95
59F0  FE 46 35 40 10 8E 00 04 :5B
59F8  A6 A9 54 BA E6 A9 54 BF :FF
5A00  DD 11 86 13 BD 6F 6C 31 :50
5A08  3F 26 ED DC A6 DD 11 86 :48
5A10  14 BD 6F 6C DC AA DD 11 :20
5A18  86 14 BD 6F 6C 0F 23 16 :7A
5A20  FD 1C BD 6B 98 96 20 81 :10
5A28  34 27 06 81 36 27 06 20 :65
5A30  1E 0A A0 20 04 0C A0 0C :A4
5A38  93 96 A0 81 04 25 0C 81 :00
5A40  34 24 02 20 0A 86 34 97 :D5
5A48  A0 20 04 86 04 97 A0 DC :61
5A50  A0 DD 16 DC A2 DD 11 86 :85
5A58  01 BD 6F 6C 96 A0 97 A2 :08
5A60  39 96 22 10 27 00 C1 1A :03
5A68  40 96 23 26 2D 03 23 96 :07
5A70  13 26 0F 96 A0 4C D6 A1 :41
5A78  DD A4 DD A6 0F F4 0C F8 :0B
5A80  20 17 96 A0 4A D6 A1 CB :F9
5A88  08 DD A4 DD A6 0F B0 8B :56
5A90  04 DD A8 DD AA 0F F4 0F :22
5A98  F8 CE 54 A4 A6 C8 50 26 :A2
5AA0  79 A6 41 90 25 81 07 23 :C0
5AA8  64 A7 41 EC C4 DD 16 EC :DB
5AB0  42 DD 11 86 03 BD 6F 6C :51
5AB8  EC C4 ED 42 D6 21 1D 1F :12
5AC0  02 A6 A9 54 80 26 14 A6 :05
5AC8  A9 53 D0 26 0E A6 A9 54 :A3
5AD0  30 A0 C4 81 02 23 0A 81 :C5
5AD8  FD 24 06 31 3F 26 E2 20 :BF
5AE0  39 A6 41 A0 A9 54 40 81 :7E
5AE8  0A 22 F0 0A 24 BD 5C 53 :B6
5AF0  6C A9 53 D0 6C A9 54 80 :21
5AF8  6C C8 50 96 13 27 06 96 :F0
5B00  F4 94 F8 27 15 0F 22 1C :09
5B08  BF 0F 23 20 1B EC C4 DD :B9
5B10  11 86 14 BD 6F 6C 0F 22 :74
5B18  1C BF 96 13 27 0A 33 44 :2C
5B20  11 83 54 AC 10 2D FF 74 :44
5B28  39 D6 21 1D 1F 02 A6 A9 :BD
5B30  54 80 26 10 A6 A9 54 30 :DD
5B38  90 A0 81 03 2E 06 81 FD :66
5B40  2D 02 20 06 31 3F 26 E6 :D1
5B48  20 30 96 A1 A0 A9 54 40 :64
5B50  81 0B 2E F0 81 F7 2D EC :3B
5B58  6C A9 53 D0 6C A9 54 80 :21
5B60  0A 24 A6 A9 54 50 E6 A9 :B0
5B68  54 60 DD 11 A6 A9 54 E0 :25
5B70  97 14 86 0E BD 6F 6C BD :94
5B78  5D 7F 39 96 2C 10 26 00 :0D
5B80  D1 BD 5D F4 81 C0 25 48 :8D
5B88  86 03 BD 5E 16 1F 89 5C :BE
5B90  1D 1F 02 A6 A9 54 C4 27 :CC
5B98  37 86 07 BD 5E 16 1F 89 :9D
5BA0  5C 1D 1F 03 A6 C9 54 80 :DE
5BA8  26 26 A6 C9 53 C0 26 20 :14
5BB0  A6 C9 53 D0 26 1A A6 C9 :41
5BB8  54 30 8B 01 A7 A9 54 B0 :64
5BC0  A6 C9 54 40 8B 0A A7 A9 :E8
5BC8  54 B5 6F A9 54 C4 26 00 :5F
5BD0  10 8E 00 04 A6 A9 54 C4 :09
5BD8  26 74 A6 A9 54 B5 8B 06 :83
5BE0  A7 A9 54 B5 81 C0 22 02 :BE
5BE8  20 15 6C A9 54 C4 A6 A9 :B1
5BF0  54 BA E6 A9 54 BF DD 11 :9E
5BF8  86 13 BD 6F 6C 20 4F 97 :37
5C00  17 A6 A9 54 BA E6 A9 54 :57
5C08  BF DD 11 A6 A9 54 B0 97 :97
5C10  16 86 05 BD 6F 6C 96 16 :E5
5C18  90 A0 81 04 2E 20 81 FF :83
5C20  2D 1C 96 17 81 A9 25 16 :5B
5C28  A6 A9 54 B0 E6 A9 54 B5 :EB
5C30  DD 11 86 13 BD 6F 6C 6C :8B
5C38  A9 54 C4 BD 5D 7F A6 A9 :A9
5C40  54 B0 A7 A9 54 BA A6 A9 :B1
5C48  54 B5 A7 A9 54 BF 31 3F :DC
5C50  26 82 39 CC 07 F4 BD 5D :C2
5C58  DD CC 08 1F BD 5D DD CC :93
5C60  09 1F BD 5D DD CC 00 50 :3B
5C68  BD 5D DD CC 01 00 BD 5D :DE
5C70  DD E6 A9 54 D0 86 10 3D :63
5C78  86 02 BD 5D DD CC 03 00 :4E
5C80  BD 5D DD CC 0B 50 BD 5D :38
5C88  DD CC 0C 09 BD 5D DD CC :81
5C90  0D 09 BD 5D DD EC C4 DD :9A
5C98  11 86 14 BD 6F 6C A6 A9 :92
5CA0  54 30 E6 A9 54 40 DD 11 :95
5CA8  86 08 BD 6F 6C C6 10 BD :B9
5CB0  5D CF 86 09 BD 6F 6C 86 :D9
5CB8  14 BD 6F 6C A6 A9 54 30 :7F
5CC0  E6 A9 54 40 DD 11 A6 A9 :60
5CC8  54 D0 97 14 86 0E BD 6F :8F
5CD0  6C A6 A9 54 D0 8B 01 9B :06
5CD8  77 BD 5D 0F 99 76 BD 5D :C9
5CE0  0F 97 76 BD 5D 1D 99 75 :61
5CE8  BD 5D 0F 97 75 BD 5D 1D :6C
5CF0  99 74 BD 5D 0F 97 74 BD :FE
5CF8  5D 1D 99 73 BD 5D 0F 97 :46
5D00  73 BD 5D 1D 99 72 BD 5D :CF
5D08  0F 97 72 BD 5D 22 39 19 :A6
5D10  85 F0 26 04 1C FE 20 02 :DB
5D18  1A 01 84 0F 39 34 01 4F :6B
5D20  35 81 34 76 CE 54 72 10 :04
5D28  8E 54 7A 10 8C 54 7F 22 :ED
5D30  2D A6 A0 A1 C0 2E 27 27 :50
5D38  F2 10 8E 54 7A CE 54 72 :F2
5D40  CC 48 28 DD 11 86 05 97 :4C
5D48  15 5F A6 C5 A7 A5 8B 30 :E6
5D50  97 14 0C 11 86 06 BD 6F :80
5D58  6C 5C C1 06 2D EC CC 48 :BC
5D60  14 DD 11 86 06 97 15 CE :08
5D68  54 72 5F A6 C5 8B 30 97 :E2
5D70  14 0C 11 86 06 BD 6F 6C :55
5D78  5C C1 06 2D EE 35 F6 BD :26
5D80  5F 84 BD 5F E8 DC A0 DD :40
5D88  11 86 0A BD 6F 6C C6 1F :1E
5D90  BD 5E 44 96 13 34 02 BD :FB
5D98  5D DD CC 0D 01 BD 5D DD :0B
5DA0  0F 13 96 9F 10 27 00 BA :48
5DA8  C6 04 3D CB 3A D7 11 86 :7A
5DB0  AC 97 12 86 10 BD 6F 6C :83
5DB8  35 02 97 13 0A 9F C6 FF :4F
5DC0  BD 5D CF DC A0 DD 11 86 :D9
5DC8  0B BD 6F 6C 0F 13 39 34 :32
5DD0  14 8E 02 00 30 1F 26 FC :15
5DD8  5A 26 F6 35 94 34 10 8E :11
5DE0  FD 00 A7 01 86 03 A7 84 :66
5DE8  6F 84 E7 01 86 02 A7 84 :8E
5DF0  6F 84 35 90 34 40 D6 93 :95
5DF8  1D D3 91 10 83 5C 00 24 :94
5E00  0E 10 83 57 00 23 08 1F :42
5E08  03 DD 91 A6 C4 35 C0 CC :9C
5E10  57 00 0F 93 20 F1 97 97 :38
5E18  8D DA 94 97 39 97 98 86 :80
5E20  3F BD 5E 16 91 98 24 02 :BF
5E28  20 04 90 98 20 F6 39 34 :CF
5E30  06 C6 15 8D 0F 35 06 0C :C4
5E38  22 0F 23 39 12 12 12 12 :D5
5E40  12 12 12 12 86 08 BD 5D :F0
5E48  DD CC 07 07 BD 5D DD CC :7A
5E50  0B C0 BD 5D DD CC 0C 10 :AA
5E58  BD 5D DD CC 0D 01 BD 5D :EB
5E60  DD 39 BD 6E 8F C6 FF BD :52
5E68  5D CF C6 FF BD 5D CF C6 :A0
5E70  FF BD 5D CF C6 FF BD 5D :C7
5E78  CF 10 FE 53 A0 96 CE 10 :44
5E80  26 F8 80 B6 53 A2 1F 8B :F3
5E88  39 96 13 10 26 F8 AF 86 :45
5E90  01 97 9E 0F 22 0F AF 0F :34
5E98  AE 86 07 BD 5E 16 97 99 :9C
5EA0  CC 02 02 DD 9A DD 9C 86 :46
5EA8  07 BD 6F 6C C6 08 BD 5D :87
5EB0  CF 96 AE 26 1C 96 22 26 :33
5EB8  05 BD 5A 22 20 13 96 A1 :A8
5EC0  80 04 81 04 10 25 00 85 :C3
5EC8  97 A1 BD 5A 4F 96 A1 97 :6C
5ED0  A3 96 AF 26 D2 96 9A 9B :AB
5ED8  9E 81 39 24 49 97 9A 96 :8C
5EE0  9B 9B 99 81 AC 24 3F 97 :F6
5EE8  9B DC 9A DD 16 DC 9C DD :59
5EF0  11 86 01 BD 6F 6C DC 9A :A6
5EF8  DD 9C 96 13 26 A9 96 9A :21
5F00  90 A0 81 04 23 04 81 FE :5B
5F08  25 9D 96 A1 90 9B 81 04 :A9
5F10  23 04 81 F0 25 91 8D 40 :1B
5F18  0C 13 0F 9E 86 FC 97 99 :7E
5F20  DC 9A DD A0 20 81 0C AF :4F
5F28  8D 2E 96 22 27 07 96 AE :E5
5F30  26 03 16 FF 72 8D 21 0F :6D
5F38  22 1C BF 86 B0 97 A1 97 :02
5F40  A3 DC A0 DD 11 86 0F BD :5F
5F48  6F 6C 16 F7 F1 0C AE 8D :20
5F50  07 96 AF 26 E0 16 FF 4F :B6
5F58  DC 9C DD 11 86 10 BD 6F :28
5F60  6C DC A2 DD 11 86 10 BD :2B
5F68  6F 6C 39 CC 07 F8 BD 5D :F9
5F70  DD CC 08 0F BD 5D DD D6 :8D
5F78  AD 4F BD 5D DD CC 01 00 :C0
5F80  BD 5D DD 39 86 10 97 AD :0A
5F88  96 A0 8B 02 97 A0 D6 A1 :71
5F90  DD 16 DC A2 DD 11 86 01 :E6
5F98  BD 6F 6C 96 A0 97 A2 8D :94
5FA0  CA 96 AD 8B 20 97 AD C6 :C2
5FA8  20 BD 5D CF 96 A0 80 02 :C1
5FB0  97 A0 D6 A1 DD 16 DC A2 :1F
5FB8  DD 11 86 01 BD 6F 6C 96 :A3
5FC0  A0 97 A2 8D A6 96 AD 81 :D0
5FC8  C8 25 10 CC 07 FF BD 5D :E9
5FD0  DD DC A2 DD 11 86 10 BD :9C
5FD8  6F 6C 39 96 AD 8B 05 97 :7E
5FE0  AD C6 20 BD 5D CF 20 A0 :3C
5FE8  DC A6 DD 11 86 14 BD 6F :36
5FF0  6C DC AA DD 11 86 14 BD :37
5FF8  6F 6C 0F 23 1C BF 39 00 :21
6000  34 06 C6 10 34 04 C6 03 :11
6008  A6 C0 12 AA 00 A7 80 5A :A3
6010  26 F6 30 88 4D 6A 60 26 :11
6018  ED 35 04 35 86 86 12 B7 :30
```

```
6020  C0 0A 86 AA B7 C0 0B 39 :B5
6028  86 43 B7 C0 0A 86 A4 B7 :2B
6030  C0 0B 39 CE C5 00 86 0B :28
6038  B7 C0 03 86 03 B7 C0 07 :81
6040  86 4D B7 C0 14 96 94 C6 :4E
6048  21 3D 33 CB 8D 15 C6 03 :C7
6050  34 50 17 FF AB 35 50 30 :FA
6058  89 40 00 33 C9 01 08 5A :28
6060  26 EE 39 96 92 C6 50 3D :C8
6068  D3 90 1F 01 39 17 FF AD :7F
6070  20 C1 8D B4 20 BD 17 FF :15
6078  A4 86 08 B7 C0 03 8D E3 :1C
6080  1F 12 D6 93 8E C4 00 58 :44
6088  3A 6E 94 20 16 20 34 20 :E6
6090  45 8D 10 20 2E 8D 0C 20 :E9
6098  3D 8D 28 20 39 8D 04 8D :69
60A0  22 20 33 86 03 B7 C0 07 :7C
60A8  86 4D B7 C0 14 1F 21 CE :6C
60B0  C8 18 C6 03 34 10 17 FF :03
60B8  47 35 10 30 89 40 00 5A :DF
60C0  26 F2 39 86 05 B7 C0 07 :5A
60C8  30 A9 02 7F 86 4B B7 C0 :A2
60D0  14 CE C8 60 20 DC 86 07 :93
60D8  B7 C0 07 30 A9 04 FE 86 :DF
60E0  49 B7 C0 14 CE C9 5F 20 :EA
60E8  C9 17 FF 3C 16 FF 8A 17 :D1
60F0  FF 2B 86 08 B7 C0 03 86 :B8
60F8  01 B7 C0 07 86 4F B7 C0 :CB
6100  14 CE C9 5F 17 FF 5C 30 :AC
6108  89 80 00 16 FE F2 17 FF :25
6110  17 20 DF D6 94 86 08 3D :4B
6118  CE D8 00 33 CB 86 08 B7 :E9
6120  C1 DA 86 01 B7 C1 DE 86 :FE
6128  4F B7 C1 E8 17 FF 34 1F :18
6130  12 34 40 D6 95 C5 01 27 :DE
6138  03 17 00 9B C5 02 27 09 :AC
6140  30 A9 40 00 EE 60 17 00 :7E
6148  8E C5 04 27 09 30 A9 80 :E0
6150  00 EE 60 17 00 81 35 C0 :DB
6158  C6 3F 34 04 E6 60 8E C4 :D5
6160  80 58 AE 85 86 EF A4 00 :24
6168  A7 00 30 88 50 8C BC 00 :F7
6170  24 11 86 10 AA 00 A7 00 :1C
6178  10 8E C4 80 E6 60 58 AF :2F
6180  A5 20 06 30 89 C1 80 20 :E5
6188  E9 6A 60 26 CF 35 84 17 :78
6190  FE 8B 86 0B B7 C0 03 86 :1A
6198  03 B7 C0 07 86 4D B7 C0 :CB
61A0  14 17 FE BF CE C9 67 16 :FC
61A8  FF 08 17 FE 7B 20 E3 17 :B1
61B0  FE 6B 86 10 B7 C0 03 86 :FF
61B8  05 B7 C0 07 17 FE A4 30 :6C
61C0  1F 86 4B B7 C0 14 CE C9 :12
61C8  CA 16 FE E6 17 FE 59 20 :52
61D0  E1 B6 D4 01 97 94 39 34 :04
61D8  06 C6 10 34 04 C6 03 A6 :83
61E0  C0 A7 80 5A 26 F9 30 88 :18
61E8  4D 6A 60 26 F0 35 04 35 :9B
61F0  86 17 FE 7E 8D 13 16 FE :CD
61F8  74 17 FE ED 8D 0B 16 FE :22
6200  75 17 FF 0A 8D 03 16 FE :39
6208  E6 DC 96 DD 91 96 95 97 :88
6210  94 39 39 39 8D 61 17 FE :42
6218  4A 30 89 40 00 CE C4 70 :45
6220  8D B5 DC 96 DD 91 17 FE :37
6228  3A 30 89 40 00 CE C9 5F :29
6230  8D A5 39 8D 42 17 FE 2B :7A
6238  30 89 40 00 CE C4 70 17 :12
6240  FF 95 DC 96 DD 91 17 FE :89
6248  1A 30 89 40 00 CE CA BA :65
6250  17 FF 84 39 86 04 8D 21 :0B
6258  17 FE 08 30 89 80 00 CE :24
6260  C4 70 17 FF 72 DC 96 DD :0B
6268  91 17 FD F7 30 89 80 00 :D5
6270  CE CA C2 17 FF 61 39 86 :90
6278  04 B7 C1 DA 86 01 B7 C1 :55
6280  DE 86 4F B7 C1 E8 39 8D :D9
6288  EE CE C4 70 17 FD D4 30 :08
6290  89 80 00 17 FF 41 39 8D :26
6298  DE CE C4 70 17 FD C4 30 :E8
62A0  89 40 00 17 FF 31 39 00 :49
62A8  00 00 00 00 00 00 00 00 :00
62B0  00 00 00 00 00 00 00 00 :00
62B8  00 00 00 00 00 00 00 00 :00
62C0  00 00 00 00 00 00 00 00 :00
62C8  00 00 00 00 00 00 00 00 :00
62D0  00 00 00 00 00 00 00 00 :00
62D8  00 00 00 00 00 00 00 00 :00
62E0  00 00 00 00 00 00 00 00 :00
62E8  00 00 00 00 00 00 00 00 :00
62F0  00 00 00 00 00 00 00 00 :00
62F8  00 00 00 00 00 00 00 00 :00
6300  34 08 86 D3 1F 8B B6 D4 :C9
6308  09 D6 8F 58 8E C4 0E AD :D3
6310  95 35 88 00 00 00 00 00 :52
6318  00 00 00 00 00 00 00 00 :00
6320  34 08 86 C4 C6 08 34 04 :8C
6328  8E 00 00 3A 6C 89 C4 30 :B1
6330  A6 89 C4 40 8B 02 A7 89 :F0
6338  C4 40 A6 89 C4 40 81 C8 :80
6340  22 02 20 05 4F A7 89 C4 :8C
6348  40 97 97 A6 89 C4 60 97 :58
6350  93 A6 89 C4 50 97 91 A6 :A4
6358  89 C4 30 97 96 4F BD C3 :79
6360  A5 A6 89 C4 40 A7 89 C4 :CC
6368  60 A6 89 C4 30 A7 89 C4 :77
6370  50 86 64 97 93 86 64 97 :E5
6378  97 BD C3 AA 96 A0 81 31 :A9
6380  27 04 0C A1 20 02 0A A1 :A5
6388  96 A2 97 91 96 A1 97 96 :C4
6390  86 01 BD C3 A5 96 A1 97 :7A
6398  A2 6A 60 26 13 12 86 07 :44
63A0  BD C3 A5 20 10 97 8F 17 :92
63A8  FF 56 B6 D4 01 97 A0 39 :50
63B0  E6 60 16 FF 73 35 04 16 :1D
63B8  FF 6A 00 00 00 00 00 00 :69
63C0  00 00 00 00 00 00 00 00 :00
63C8  00 00 00 00 00 00 00 00 :00
63D0  00 00 00 00 00 00 00 00 :00
63D8  00 00 00 00 00 00 00 00 :00
63E0  00 00 00 00 00 00 00 00 :00
63E8  00 00 00 00 00 00 00 00 :00
63F0  00 00 00 00 00 00 00 00 :00
63F8  00 00 00 00 00 00 00 00 :00
6400  C0 8B C0 8D C0 8F C0 91 :38
6408  C0 95 C0 99 C0 9D C1 F1 :BD
6410  C1 F9 C2 01 C2 14 C2 33 :48
6418  C2 54 C1 13 C1 58 C1 8F :53
6420  C1 AA C1 AF C1 CC C1 D1 :FA
6428  C0 6D C0 72 C0 76 C0 E9 :3E
6430  C0 EF C1 0E C2 87 C2 97 :20
6438  22 22 22 22 22 22 22 22 :10
6440  22 22 22 B6 B8 76 78 78 :3A
6448  7A 2D 2A 2A 2B 2B 2B 2B :A7
6450  2F 7C 97 6C 42 1F E2 2B :1C
6458  FC 2D 24 01 2B 2B 19 25 :E2
6460  00 7C 5C B6 B8 76 78 78 :AC
6468  7A 2D 2A 2A 2B 2B 2B 2B :A7
6470  00 00 00 00 00 00 00 00 :00
6478  00 00 00 00 00 00 00 00 :00
6480  8D 41 A7 CF B2 56 BB B2 :B9
6488  A2 C4 81 09 A1 3E 9D 22 :8E
6490  BD 46 91 88 87 3D A3 19 :9C
6498  87 AD B6 7B A6 2A A7 30 :0C
64A0  A6 9A 94 16 A0 30 96 46 :96
64A8  A9 B6 95 42 8C 5A 9A A6 :5C
64B0  9E CF 94 5A A9 62 AA D5 :E5
64B8  89 38 98 CE 93 8E A7 81 :70
64C0  81 E2 AC 81 AC E5 AF 86 :56
64C8  95 72 86 B7 A5 DD AD A5 :18
64D0  8C A4 A6 D2 90 1F AE 18 :1D
64D8  8E 2E 93 E0 96 B0 90 E7 :EC
64E0  8C 56 84 E8 92 86 9E 29 :2D
64E8  9E A5 A7 CD 96 A1 88 41 :B7
64F0  91 61 9E FA A0 9B 99 0C :6A
64F8  A8 A9 86 A7 85 D0 8A 7B :D8
6500  00 00 00 00 00 00 00 00 :00
6508  00 01 00 00 07 C0 00 01 :C9
6510  00 00 00 00 00 00 00 00 :00
6518  00 00 00 00 00 00 00 00 :00
6520  00 00 F8 00 01 F8 00 00 :F1
6528  24 00 07 CF 00 0F 83 3E :CA
6530  00 7C 7F 3E 78 3E 60 27 :76
6538  C0 1F 0F E0 3F 87 C0 1F :73
6540  00 00 00 0F 00 00 18 00 :27
6548  00 18 00 00 00 00 00 00 :18
6550  00 00 00 00 00 00 00 00 :00
6558  00 00 00 00 00 00 00 00 :00
6560  00 00 00 00 00 00 00 00 :00
6568  00 00 00 00 00 00 00 00 :00
6570  00 00 00 00 00 00 00 00 :00
6578  00 00 E0 1C 00 00 00 00 :FC
6580  00 00 00 00 00 00 00 00 :00
6588  00 00 00 00 00 00 00 00 :00
6590  00 30 00 00 C0 00 00 C0 :B0
6598  00 00 00 00 00 00 00 00 :00
65A0  00 00 00 00 00 00 00 00 :00
65A8  02 82 A0 0A 8A A8 2A AA :34
65B0  AA 2A AA AA 2A AA AA 2A :D0
65B8  AA AA 2A AA AA 0A AA A8 :2E
65C0  02 A2 A0 00 00 00 00 00 :44
65C8  00 00 00 00 00 00 00 00 :00
65D0  00 00 00 00 00 00 00 00 :00
65D8  00 00 00 00 00 00 00 00 :00
65E0  00 00 00 00 00 00 00 00 :00
65E8  00 00 00 00 00 00 00 00 :00
65F0  00 00 00 00 00 00 00 00 :00
65F8  00 00 00 00 00 00 00 00 :00
6600  00 00 00 00 00 00 00 00 :00
6608  00 08 00 01 FF C0 0F C1 :98
6610  F8 1F FF FC 3F FF FE 3F :8D
6618  FF FE 3F FF FE 1F FF FC :53
6620  0F FF F8 01 FF C0 00 08 :CE
6628  00 00 F8 00 01 F8 00 00 :F1
6630  24 00 07 CF 00 0F 83 3E :CA
6638  00 7C 7F 3E 78 3E 60 27 :76
6640  C0 1F 0F E0 3F 87 C0 1F :73
6648  00 00 01 F0 C0 0F E7 F8 :9F
6650  1F 67 7C 7F AA 7F 7F FF :28
6658  FF FF FF FF 7F FF FF 7F :F8
6660  FF FF 1F FF FC 0F FF F8 :1E
6668  01 FF 00 00 00 00 00 00 :00
6670  00 00 00 00 00 00 00 00 :00
6678  00 00 00 00 FC 0F 83 FF :8D
6680  3F E7 FF 7F F3 FF 3F E0 :B5
6688  FC 0F 80 00 00 00 00 00 :8B
6690  00 00 00 3F F8 01 FF F8 :2F
6698  03 FF F0 07 FF F0 07 FF :EE
66A0  E0 1F FF C0 1F FC 00 00 :D9
66A8  00 00 00 00 00 00 00 00 :00
66B0  03 C7 E0 1F DF FC 3F FF :E2
66B8  FE 7F FF FF 7F FF FF 7F :77
66C0  FF FF 3F FF FE 1F FF FC :54
66C8  03 E7 E0 00 00 00 00 00 :CA
66D0  3F 00 0F FF 03 FF FC FF :4A
66D8  3F FF 7F F3 FF 3F F3 FF :E0
66E0  1F FC FE 0F FF FE 07 F3 :1F
66E8  FC 01 FF F8 00 3F E0 00 :13
66F0  00 0F 00 00 FF 00 0F C7 :E4
66F8  03 FF 0E 7F F0 1E 60 00 :FD
6700  38 00 03 F0 00 FF C0 7F :69
6708  FF 00 FF C0 00 30 00 00 :EE
6710  00 00 00 01 55 40 05 41 :DC
6718  50 15 55 54 17 D5 54 15 :63
6720  55 54 15 55 54 15 55 54 :25
6728  05 55 50 01 55 40 00 00 :40
6730  00 00 00 00 00 03 1E 00 :21
6738  03 FE 00 00 FC 00 00 00 :FD
6740  00 00 00 00 00 00 00 00 :00
6748  00 00 00 00 00 00 00 00 :00
6750  00 00 00 0F 00 00 18 00 :27
6758  00 18 00 04 00 00 1F 00 :3B
6760  00 04 00 00 00 00 00 00 :04
6768  00 00 00 00 00 00 00 00 :00
6770  00 00 00 1C 00 00 17 80 :B3
6778  00 08 F0 00 08 18 00 04 :1C
6780  06 00 04 03 00 00 00 00 :0D
6788  00 00 E0 1C 00 00 00 00 :FC
6790  00 00 00 00 00 00 00 00 :00
6798  00 00 00 3F F8 01 FF F8 :2F
67A0  03 FF F0 07 FF F0 07 FF :EE
67A8  E0 1F FF C0 1F FC 00 00 :D9
67B0  00 00 00 00 00 00 00 00 :00
67B8  02 82 A0 0A 8A A8 2A AA :34
67C0  AA 2A AA AA 2A AA AA 2A :D0
67C8  AA AA 2A AA AA 0A AA A8 :2E
67D0  02 A2 A0 00 00 00 00 00 :44
67D8  00 00 00 00 00 00 00 00 :00
67E0  00 00 80 00 00 C0 00 00 :40
67E8  60 00 02 70 00 02 38 00 :0C
67F0  0C 1E 00 38 07 FF E0 00 :48
67F8  00 0F 00 00 FF 00 0F FF :1C
6800  03 FF CE 7F FF 1E 7F F0 :DB
6808  3C 00 03 F8 00 FF F0 7F :A5
6810  FF C0 FF FF 00 3F F0 00 :EC
6818  00 00 00 01 FF 80 07 81 :08
6820  E0 0F 00 F0 0F 00 F0 0F :ED
6828  E7 F0 00 3C 00 00 00 00 :13
6830  00 3C 00 00 00 00 00 7E :BA
6838  00 00 E7 00 C0 FF 03 C0 :69
6840  18 03 C0 00 03 E0 00 07 :C5
6848  00 00 00 01 FF 80 07 FF :86
6850  E0 0F E7 F0 3F FF FC 0F :0F
6858  FF F0 00 FF 00 00 3C 00 :2A
```

```
6860   00 00 00 00 00 01 FF 81 :81
6868   FF 80 07 81 E7 81 E0 0F :5E
6870   00 EF 00 F0 0F 00 F7 00 :E5
6878   F0 0F E7 EF E7 F0 00 3C :E8
6880   00 3C 00 00 00 00 00 00 :3C
6888   00 3C 00 3C 00 00 00 3C :B4
6890   00 00 00 7E 18 7E E0 00 :F4
6898   E7 EF E7 00 C0 FF 08 FF :83
68A0   03 C0 18 10 18 03 C0 00 :C6
68A8   18 00 03 E0 00 18 00 07 :1A
68B0   00 00 00 00 00 01 FF 81 :81
68B8   FF 80 07 FF E7 FF E0 0F :5A
68C0   E7 EF E7 F0 3F FF F7 FF :E1
68C8   FC 0F FF EF FF F0 00 FF :E7
68D0   00 FF 00 00 3C 00 3C 00 :77
68D8   00 00 00 00 00 00 00 00 :00
68E0   00 00 00 00 00 00 00 00 :00
68E8   00 00 00 00 00 00 00 00 :00
68F0   00 00 00 00 00 00 00 00 :00
68F8   00 00 00 00 00 00 00 00 :00
6900   00 00 00 00 00 00 00 00 :00
6908   00 00 00 00 00 00 00 00 :00
6910   00 00 00 00 00 00 00 00 :00
6918   00 00 00 00 00 00 00 00 :00
6920   00 00 00 00 00 00 00 00 :00
6928   00 00 00 00 00 00 00 00 :00
6930   00 00 00 00 00 00 00 00 :00
6938   00 00 00 00 00 00 00 00 :00
6940   00 00 00 00 00 00 00 00 :00
6948   00 00 00 00 00 00 00 00 :00
6950   00 00 00 00 00 00 00 00 :00
6958   00 00 00 00 00 00 00 18 :18
6960   18 18 3C 00 00 00 00 80 :EC
6968   00 38 60 60 F8 38 93 10 :CB
6970   1B 9C 10 0C 00 20 1C 00 :0F
6978   00 F8 00 20 04 01 18 0F :44
6980   3C FE 3E 38 07 F8 70 00 :1F
6988   80 00 38 60 60 F8 38 F3 :9B
6990   F0 1F FF F0 0F FF E0 1F :0B
6998   FF FC FF FF E0 07 FF F8 :D7
69A0   0F FC FE 3E 38 07 F8 70 :EE
69A8   00 80 00 38 60 60 F8 38 :A8
69B0   F3 F0 1F FF 30 0F BC 60 :5C
69B8   1E 00 7C FF 80 E0 07 FB :FB
69C0   F8 0F FC FE 3E 38 07 F8 :76
69C8   70 00 00 00 00 00 00 00 :70
69D0   00 08 00 00 00 00 08 00 :10
69D8   00 00 00 1C 00 00 00 00 :1C
69E0   1E 00 00 0F 00 8F E0 00 :9C
69E8   00 F0 FF FF F0 00 0F FF :EC
69F0   F8 00 00 07 FF C0 00 00 :BE
69F8   07 FF F8 00 00 0E 7E 18 :A2
6A00   00 00 18 1C 0C 00 00 40 :80
6A08   08 02 00 00 00 00 00 00 :0A
6A10   00 08 00 00 00 00 00 00 :08
6A18   00 00 00 00 1C E0 71 00 :6D
6A20   00 FF F8 FC 00 23 FF FF :14
6A28   FC 03 FF FF FF FC FF FF :F6
6A30   FF FF F0 1F FF FF FF FF :09
6A38   1F FF FF FF 1F FF FF FF :38
6A40   FF FC 0F FF FF FF F8 0F :0E
6A48   3F FF FF FE 07 FF FF FF :3F
6A50   F8 07 FF FF FF E0 03 FF :DE
6A58   FF FF C0 01 FF FF FF C0 :7C
6A60   03 1F DC F8 78 0C 0F 1C :A5
6A68   00 0F 00 00 00 E0 00 00 :EF
6A70   00 FF F8 00 00 03 C9 FB :BE
6A78   C0 03 CF DC 47 E0 0F E3 :87
6A80   DE 0F F0 0F 00 1F E0 00 :EB
6A88   00 F0 FF FF F0 1F 0F FF :0B
6A90   F8 0F 0F C7 FF C3 FC 00 :9B
6A98   07 FF FB FE 07 8E 7E 1B :2D
6AA0   F8 07 18 1C 0D E0 03 21 :44
6AA8   C9 E2 C0 00 07 EB F8 00 :55
6AB0   00 1F C8 F8 00 00 0F 00 :EE
6AB8   00 00 33 33 33 33 33 33 :32
6AC0   33 33 3C 18 18 18 00 00 :EA
6AC8   00 00 00 00 00 00 00 00 :00
6AD0   00 00 00 00 00 00 00 00 :00
6AD8   00 00 00 00 00 00 00 00 :00
6AE0   00 00 00 00 00 00 00 00 :00
6AE8   00 00 00 00 00 00 00 00 :00
6AF0   00 00 00 00 00 00 00 00 :00
6AF8   00 00 00 00 00 00 00 00 :00
6B00   34 30 BD 6F 47 8E FC 80 :E1
6B08   86 15 A7 02 6F 03 6F 04 :29
6B10   CC 01 E1 ED 05 CC 00 32 :9E
6B18   ED 07 CC 02 7F ED 09 CC :03
6B20   00 AA ED 0B 86 02 A7 0D :DE
6B28   7F FD 05 5F 34 04 C6 32 :10
6B30   A6 E4 81 08 24 5E 34 04 :CD
6B38   5F 9E 1C 31 88 30 A6 61 :09
6B40   A1 A0 27 0D E1 84 27 03 :04
6B48   5C 20 F3 35 04 6C E4 20 :18
6B50   DF 35 04 D7 12 86 41 97 :5F
6B58   11 A6 E4 97 14 86 0D BD :96
6B60   6F 6C 86 3D 97 14 86 47 :16
6B68   97 15 86 45 97 11 86 06 :AB
6B70   BD 6F 6C 86 47 97 11 A6 :B3
6B78   E4 4C 8B 30 97 14 86 06 :22
6B80   BD 6F 6C 0C 11 86 30 97 :02
6B88   14 86 06 BD 6F 6C CB 0F :12
6B90   6C E4 20 9C 35 B4 00 00 :F5
6B98   7D 6B DC 27 40 34 76 4F :24
6BA0   B7 6B D9 8E 6B D7 BD F1 :79
6BA8   7D B6 6B DB C6 34 81 32 :26
6BB0   25 08 C6 36 81 C8 24 02 :98
6BB8   C6 32 D7 20 96 22 26 15 :E2
6BC0   86 03 B7 6B D9 8E 6B D7 :54
6BC8   BD F1 7D B6 6B DB 81 32 :DA
6BD0   24 03 BD 5E 2F 35 F6 00 :9C
6BD8   00 03 48 FF 00 34 03 20 :A1
6BE0   16 34 03 1A 10 86 01 B7 :B5
6BE8   FD 02 B6 FD 01 B6 FD 03 :69
6BF0   85 01 26 F9 7F FD 02 B6 :D9
6BF8   FD 01 97 20 35 83 00 00 :6D
6C00   BD 6F 47 CE FC 82 86 2A :6F
6C08   A7 C0 86 06 A7 C0 86 04 :E4
6C10   A7 C0 86 41 A7 C0 86 55 :70
6C18   A7 C0 86 54 A7 C0 86 4F :7D
6C20   A7 C4 7F FD 05 39 34 56 :AF
6C28   CE 53 F0 8E FC 80 BD 6F :47
6C30   47 86 15 A7 02 A6 48 A7 :20
6C38   03 A6 49 A7 04 EC C4 ED :3A
6C40   05 EC 42 ED 07 EC 44 ED :44
6C48   09 EC 46 ED 0B A6 4A A7 :CA
6C50   0D 7F FD 05 35 D6 34 76 :43
6C58   CE 53 F0 10 8E FC 82 BD :EA
6C60   6F 47 86 19 A7 A0 A6 48 :8A
6C68   A7 A0 A6 49 A7 A0 6F A0 :8C
6C70   A6 4B A7 A0 A6 4C A7 A0 :71
6C78   EC C4 ED A1 EC 42 ED A1 :FA
6C80   E6 80 E7 A0 A6 80 A7 A0 :5A
6C88   5A 26 F9 7F FD 05 35 F6 :25
6C90   CC 00 00 FD 53 F0 CC 00 :D8
6C98   42 FD 53 F2 CC 02 7F FD :CE
6CA0   53 F4 CC 00 72 FD 53 F6 :CB
6CA8   7F 53 F8 86 02 B7 53 FA :56
6CB0   BD 6C 26 CC 00 B0 FD 53 :1B
6CB8   F2 CC 00 C7 FD 53 F6 BD :88
6CC0   6C 26 CC 00 00 FD 53 F0 :9E
6CC8   FD 53 F2 CC 02 7F FD 53 :DF
6CD0   F4 CC 00 41 FD 53 F6 86 :CD
6CD8   02 B7 53 F8 86 02 B7 53 :96
6CE0   FA BD 6C 26 CC 00 73 FD :85
6CE8   53 F2 CC 02 7F FD 53 F4 :D6
6CF0   CC 00 B4 FD 53 F6 86 01 :4D
6CF8   B7 53 F8 BD 6C 26 CC 00 :1D
6D00   0A FD 53 F0 CC 00 78 FD :8B
6D08   53 F2 CC 02 76 FD 53 F4 :CD
6D10   CC 00 AF FD 53 F6 86 07 :4E
6D18   B7 53 F8 BD 6C 26 CC 00 :1D
6D20   05 FD 53 F2 CC 00 3C FD :4C
6D28   53 F6 86 07 B7 53 F8 BD :95
6D30   6C 26 F6 51 00 86 10 3D :AC
6D38   34 06 CC 01 40 A3 E4 FD :CB
6D40   53 F0 35 06 CC 00 50 FD :97
6D48   53 F2 8E 51 00 86 04 B7 :65
6D50   53 FB 86 02 B7 53 FC 86 :62
6D58   07 B7 53 F8 BD 6C 56 CC :54
6D60   00 14 FD 53 F0 CC 00 20 :40
6D68   FD 53 F2 8E 51 50 86 04 :FB
6D70   B7 53 FB 86 01 B7 53 FC :92
6D78   86 01 B7 53 F8 BD 6C 56 :08
6D80   CC 00 50 FD 53 F0 CC 00 :28
6D88   82 FD 53 F2 8E 51 25 86 :4E
6D90   02 B7 53 FB 86 01 B7 53 :98
6D98   FC 7F 53 F8 BD 6C 56 CC :11
6DA0   00 50 FD 53 F0 CC 00 A0 :FC
6DA8   FD 53 F2 8E 51 44 86 02 :ED
6DB0   B7 53 F8 BD 6C 56 CC 00 :4D
6DB8   00 FD 53 F0 CC 00 B9 FD :C2
6DC0   53 F2 8E 51 64 86 02 B7 :C7
6DC8   53 FB 86 01 B7 53 FC 86 :61
6DD0   05 B7 53 F8 BD 6C 56 CC :52
6DD8   00 FC FD 53 F0 CC 00 B9 :C1
6DE0   FD 53 F2 8E 51 6F 86 02 :18
6DE8   B7 53 FB 86 01 B7 53 FC :92
6DF0   86 07 B7 53 F8 BD 6C 56 :0E
6DF8   CC 01 2C FD 53 F0 CC 00 :05
6E00   B9 FD 53 F2 8E 51 15 BD :AC
6E08   6C 56 39 CC 00 00 FD 53 :17
6E10   F0 FD 53 F2 CC 02 7F FD :7C
6E18   53 F4 CC 00 C7 FD 53 F6 :20
6E20   7F 53 F8 BD 6C 26 CC 01 :E6
6E28   E0 FD 53 F4 86 05 B7 53 :B9
6E30   F8 86 01 B7 53 FA BD 6C :AC
6E38   26 CC 02 08 FD 53 F0 CC :08
6E40   00 14 FD 53 F2 8E 51 73 :A8
6E48   86 01 B7 53 FB 86 01 B7 :CA
6E50   53 FC 86 07 B7 53 F8 BD :9B
6E58   6C 56 CC 01 F0 FD 53 F0 :BF
6E60   CC 00 28 FD 53 F2 8E 51 :15
6E68   79 BD 6C 56 CE 54 7A 86 :1A
6E70   48 97 11 86 28 97 12 86 :CD
6E78   05 97 15 A6 C0 8B 30 97 :69
6E80   14 0C 11 86 06 BD 6F 6C :55
6E88   11 83 54 7F 23 ED 39 CC :7C
6E90   00 50 FD 53 F0 CC 00 64 :C0
6E98   FD 53 F2 86 02 B7 53 F8 :CC
6EA0   86 04 B7 53 FB 86 02 B7 :CE
6EA8   53 FC 8E 51 82 BD 6C 56 :2F
6EB0   39 00 00 00 00 00 00 00 :39
6EB8   BD 50 DE 0F 22 1C BF 39 :30
6EC0   BD 6B E1 7F 6B DC 96 20 :85
6EC8   81 53 27 0D 81 42 10 27 :02
6ED0   E7 E9 81 4A 26 EA 7C 6B :92
6ED8   DC 7E 57 1D 1A 40 7D 6B :10
6EE0   DC 26 13 34 06 C6 15 BD :E7
6EE8   5E 44 A6 62 8A 40 A7 62 :7D
6EF0   35 06 0C 22 0F 23 3B 00 :D6
6EF8   00 00 00 00 00 00 00 00 :00
6F00   34 72 10 BF 6F 45 BC 6F :54
6F08   45 22 23 FF 6F 42 A6 80 :60
6F10   B7 6F 40 34 50 BD 6F 47 :5D
6F18   8E 6F 30 10 8E 6F 44 CE :4C
6F20   FC 80 BD 6F 5E 35 50 33 :BE
6F28   41 7F FD 05 20 D8 35 F2 :E1
6F30   00 00 3F 59 41 4D 41 55 :BC
6F38   43 48 49 93 D3 8F 90 86 :DF
6F40   00 B7 C8 17 39 38 62 34 :9D
6F48   02 B6 FD 05 2B FB 86 80 :E6
6F50   B7 FD 05 B6 FD 05 2A FB :96
6F58   35 82 7F FD 05 39 34 70 :15
6F60   AC 62 22 06 A6 80 A7 C0 :C3
6F68   20 F6 35 F0 34 56 B7 54 :D0
6F70   0F BD 6F 47 8E 54 00 CE :32
6F78   FC 80 11 83 FC 96 22 06 :CA
6F80   EC 81 ED C1 20 F4 7F FD :AB
6F88   05 35 D6 34 02 86 0C BD :95
6F90   6F 6C BD 6F 47 B6 FC 94 :94
6F98   97 20 86 80 B7 FC 80 7F :6F
6FA0   FD 05 35 82 34 06 DC 2E :FD
6FA8   C3 00 01 DD 2E 10 83 FD :5F
6FB0   00 24 1C A6 9F 54 2E 81 :88
6FB8   60 25 08 81 A0 22 0A 0F :E9
6FC0   2D 35 86 86 01 97 2D 35 :68
6FC8   86 86 FF 97 2D 35 86 CC :56
6FD0   82 CC DD 2E 20 DD 3B FC :8D
6FD8   FF F6 FD 53 16 39 CC 6F :CF
6FE0   D6 FD FF F6 39 FC 53 16 :66
6FE8   FD FF F6 39 00 00 00 00 :2B
6FF0   00 20 22 00 7F 72 13 8E :D4
```

構造化プログラミングの理論

滝本 康夫

構造化プログラミングとは，良いプログラムを作るための技法ではない。正しくは，良いプログラムを作るための考え方である。今回は，構造化プログラミングの理論的側面について考えてみよう。

構造化プログラミングの位置づけ

良いプログラムを設計しようとするときに，BASICやPascalなどの文法をマスタすることは，まったく意味がない。良いプログラムを実現させるという行為は，特定の言語とは独立だからである。

たしかにサポートツールは，プログラム構造を望ましくするために有効である。しかし，有効な構造をサポートするためにツール（構造化言語を含む）が使用されるのであり，逆にツールを使用すれば自然に構造化が実現されるというわけでは決してない。だから，たとえBASIC言語しか使用できない場合であっても，正しい考え方をすればなんら不都合なく，構造化プログラミングを行うことができるわけである。

現実にアセンブラであっても構造化手法を用いているくらいだから，BASICはそれに比較すれば，ずっと便利な機能を備えている。

プログラミングという行為そのものがそうであるように，構造化プログラミングという行為もまた，きわめて人間依存の度合いの高い作業といえるだろう。

さて，その構造化プログラミングとは，どのような項目から成るのだろうか。

思想的なとらえ方をすると，前回述べたように，

①わかりやすいプログラム設計

②わかりやすいプログラムコーディング

③わかりやすいプログラムテスト

となる。ところが，これが実際にどのような技法で展開されるかというと，ちょっとつかみがたいところがある。

①トップダウンによる設計

②プログラムのモジュール化

③基本3構造によるコーディング

入門者への説明としてはこの程度でも充分である。中には③だけでもって構造化手法という豪胆な人もいるくらいで，これは望ましい説明といえよう。だが，ちょっとしつこく考えてみると，これらの説明だけでは完全とはいえない。

なぜなら，トップダウン法というのは，ボトムアップ法とともに，構造化とは直接には関係ない別のところから出てきた手法である。また，モジュール化プログラミングも独立の手法である。

構造化ではモジュールという概念を使用するが，ただ単に使用するのではなく，これに階層構造というテクニックを付加することになる。

さらに段階的詳細化も，構造化プログラミングでは重要な手法である。

一方，階層構造，モジュール化，各ステップごとの洗練化を統合してトップダウン設計，と述べている学者もいる。例のDijkstra教授でさえ，構造化プログラムといわずに，

「階層プログラム構造に限ることを学ぶのが早いほど，より速く進歩できる」

などと述べている。

べつにその解釈について，あれこれ判断を下すのが目的ではないから，この程度にするが，要するに，

——構造化プログラミングやその他のソフトウェア工学技法，あるいはまたその技法により生ずる利点は，多くの人にとって漠然としたわかりづらいもの

とは言えそうである。

1950年代のプログラミングは，家内工業プログラミングと呼ばれていて，限定されたわずかの人々が職人芸丸だしでコードを作り，作成者がそのままコードを保守してきた。当然，柔軟性も拡張性もない，はなはだ読みにくいコードが作成されたが，時代の要求に対しては何も問題がなかった。

1960年代になってからは，大きなプログラム，変更と拡張作業，共同開発などが行われる重工業的プログラミングとなり，いろいろな問題点が発生してきた。

1970年代には，それらの問題点が重大なものとなり，その改善行為が社会的意義のあるものとしてとらえられ始めた。それがデータ構造であり，階層構造であり，モジュール化，トップダウン，さらにはHIPOやNSチャートなどの発想法，文書化法である。

これらに共通していることは，プログラミング行為を何らかの型（＝構造）に当てはめようとしていることである。さまざまな学者や研究者が，それぞれの閉ざされた世界で独自に見い出した手法も，たどってみれば根は一つだった，というわけだ。

現在では，これらのさまざまなアプローチを総称して，構造化プログラミングと呼んでいるようである。

筆者自身，長い間構造化プログラミング

の定義には悩んできたが，上述のように断言している書物を見たときには，ようやく胸のつかえがとれたような気がした。

反面，わかってしまうと，重要なのはプログラミング行為そのものであって，意味にこだわるのは意味のないことだと，まるでこのダジャレのような軽さしか感じない。不思議なものだ。

構 造 化 設 計

システム開発は，通常，

①要件分析

②仕様決定

③設計

④コーディング

⑤テスト

⑥運用

という手順で行われる。機能分析とか，内部設計，外部設計という用語を用いる人もいるが，いわんとするところは同じである。

この中で，前の工程にあるものほど，ミスを犯したときの波及が大きい。バグとはプログラムコードの中だけに存在するのではない。

たとえば項目数1万件のデータ処理システムを，開発時に誤って，1000件のサイズに設計してしまったら，その修正費用は膨大なものになるだろう。だが，これに対して，コーディング作業中にある1ヵ所だけ単に

WHILE CT<10000

を，

WHILE CT<1000

と間違えた，という場合なら(前者に比べ)修正コストは安価なものである。

さてこれらの工程の中で，バグの入り込む可能性の高いものは，何といっても設計工程であろう。

設計エラーはコーディングエラーに比べて，調査するのに時間がかかり，修正にも時間がかかる。コスト的な見方をすると，システム開発作業時の大部分のエラーは，設計エラーであるといっても過言ではない。

だから，システム開発中の早い時期に設計エラーを見つけなくてはならないし，またミスのはいり込む可能性の少ない設計をしなくてはならない。

これら設計の手法を改善するための方法論を統合したものが構造化設計と呼ばれているものである。構造化設計の目的は，プログラム内容をわかりやすくすることであり，

①設計の過程を体系化する

②複雑さを少なくする

③問題解決が効率よく進むような仕事の段取りを設定する

という効果を狙うことである。そのための具体的手段として，

①プログラムの分割

②独立性を高める

③階層構造化を行う

などの手法が用いられる。

また，構造化設計を高品質に行わせるために，一度に決定する項目を1つだけにするといった段階的詳細化が用いられるのである。これはいかに重大な決定であっても，その内容は単純な決定の組み合せによるという考え方に基づいている。

構造化設計を支えるもう1つの概念は，トップダウン法である。トップダウン法自体もまた，段階的詳細化の考え方を採用している。が，このあたりの思考法はいろいろからみあっていて判然としない。

トップダウン法とは，システムの機能をより小さい機能に分割し，この作業を繰り返すことにより，システムは次第に細分化された部分機能に分解されていく，というものである。これに対し，具体的な処理部分を先に作り，これを組み合せ，積み上げていくことによりシステムを完成させるボトムアップ法がある。

両者はまったく逆の考え方であるが，その使用法については誤解されている面もある。ある雑誌では次のように書かれていた。

――人間の心理としては，いちばん難しそうなところは自信がないから先にやってみて，もしうまくいったら次を考えてみよう，というのが人情のようです。これをプログラムの設計でみれば，まずプログラムの細部を考えてから全体を仕上げる，というボトムアップの手法になると思います。実はこの方法は，非能率的！なのです。プログラムの設計はあくまでもトップダウン思考によるのがよいのです。

この筆者は，たぶんトップダウン法の有効性を強調したかったのだろう（気持ちはよくわかる）けど，ちょっと言いすぎの感がある。それは「あくまでトップダウン思考によるものがよいのです」のくだりだ。

たしかに通常は，トップダウン法の方が有効である。しかし，断定してしまえるほど完全ではない。

トップダウン法に適しているのは，設計チームの能力からして，あらゆる低レベルモジュールの作成が可能な見通しがたっている場合であり，これはほぼすべてのビジネス用アプリケーションにあてはまる。一方，非常に高度な技法を要求されるシステムを開発する場合には，困難と思われるモジュールの実現性を先に検討し，だめなら計画を修正することのできるボトムアップ法が，やはり不可欠である。また，現存システムを拡張する，流用するというようなときも，ボトムアップ法が向いている。

ソフトウェア上の手法などは断定すると皆まちがいである。必ず無理を生じるものだ。難しいシステムを作る場合の実際の設計過程は，トップダウン法とボトムアップ法の組み合せを繰り返すことになる場合が多い，というのが正直なところであろう。

以下，モジュール，階層構造，独立性について簡単に説明を加える。

まず，モジュールとは何か。マイヤーズの定義や，その他の記述を参考にすると，ほぼ次のようになるだろう。モジュールとは，

①1つの独立した機能を持つ
②複数の命令群より成り，明確に境界が区切られ，固有の名前が付いている
③1つの入口と1つの出口を持つ
④別々にコンパイルし，テストできる
⑤他のモュールからの呼び出しが可能

これらのことによりモジュールとは，ソフトウェアシステムの行う各種の働きの基本動作を実行するものと考えてよいだろう。

たとえば構造化言語における，外部手続き，関数，副プログラム，サブルーチン(外部に置くもの）などは，すべてモジュールという定義にかなう。言葉をかえると，要するにコンパイルやリンク時に外部から取り込まれるプログラム単位，と述べることができる。

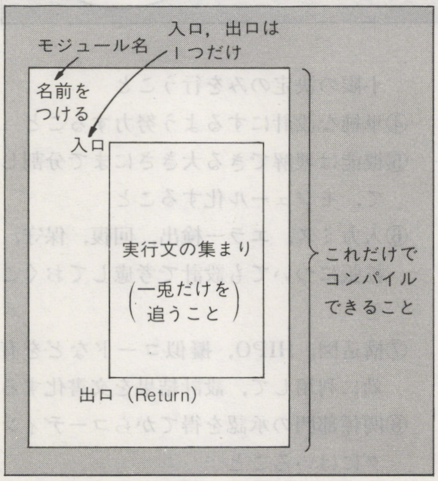
だが，ここで疑問が1つ生じてくる。読者もお気づきだろう。

それは，たとえばPascalの内部手続きや，身近な例ではBASICにおけるサブルーチンはモジュールではないのか，ということである。

BASICのサブルーチンは，メインルーチンと一体になって同じファイル名の中に存在するから，CHAIN文などでディスクから引き込む形にしないかぎり，モジュールの定義からはずれる。

しかし現実には，これらのものもモジュールと呼んでいる。モジュールの定義もつきつめると曖昧なものであり，精神的にモジュールと呼ぶ，ということも許されている。逆にそうしないと，プログラムは実体とかけ離れたものになってしまうのだ。

▼図1　モジュールとは



もっと柔軟な解釈例では，定義⑤他のモジュールからの呼び出しが可能，という項目についてもいえる。この定義によれば，分割コンパイルの件は別にして，

```
100 GOSUB 140
110 GOSUB 180
120 GOSUB 220
130 END
140 ′----------md1
150   ---
160   ---
170 RETURN
180 ′----------md2
190   ---
200   ---
210 RETURN
220 ′----------md3
230   ---
240   ---
250 RETURN
```

におけるmd1，md2，md3はモジュールであるが，

```
100 ′----------md1
110   ---
120   ---
130 ′----------md2
140   ---
150   ---
160 ′----------md3
170   ---
180   ---
190 END
```

はモジュールとはいえないことになる。後者ではGOSUB文による呼び出しが行われていないからである。まったく同じことをやっていて片方はモジュール，他方はモジュールではないというのは不都合な話だ。

現実的対応としては，モジュール化の徹底により個々の機能をすべて独立させて持つことは，管理の煩わしさの点で望ましくない。したがってそのような場合は，内部手続きや，段落づけなどで，単位機能を実現することも多い。つまり，順に実行できる独立した機能だけは，階層を除いたモジュール性のみを使用することも認められているのである。

次に，プログラムをわかりやすくするために，機能分解と同様に階層構造化の概念が必要となる。これは人間の考え方としては自然なものであり，特に理屈をならべなくとも，なるほど階層構造化は効果的である，という感想をもたれると思う。

今，図2のような機能Xがあったとする。このXとは，たとえば給与計算とか，生産報告書を出力するとかいったことである。

機能Xは，大きくXA，XB，XCの3つに分解できるとする。サブ機能XBは，さらに図3のように，XB_1，XB_2に分解できる。すると，機能Xは，図4のような形で表現できる。これは非常にわかりやすい。どの機能はどのサブ機能によりなり，どの親機能に従属しているかが，一目瞭然となる。プログラムリストを見ただけでは，この関係は不明確だ。

階層構造化はこのように，完成したものの相互関係を知る場合に有効であるが，もうひとつ重要な目的をもっている。

それは，プログラムの機能分解をやりやすくする，ということである。

たとえば給与計算のプログラムを作ろうとする場合に，いきなり基本単位のモジュール名を，次々に言える人などいないだろう。まず，給与計算という機能を，とりあえず3,4個のかたまりに分けるのが自然だ。そして，そのかたまりを，より小さな数個の機能に分けていくはずである。

階層化へのアプローチは，問題の全体像を抽象的なものから，より具体的な小さな単位へ，間違いのないように少しずつ段階的に整理していくという方法を可能にする。

モジュールに関するもうひとつの重要ポイントは，独立性である。

機能分解されたそれぞれのモジュールは，他のどこからも影響をうけない独立国でなければ，管理が難しい。何か対策を決定するのにソ連や中国のことを考慮しなければならないとしたら，その決定や変更は容易ではないはずだ。

図5を見ていただきたい。このように，A, B, C, Dという4つの機能に分解したとき，もしA,Bの中にCの影響部分が含まれていたとしたら，Cを変更するとAもBも変更する必要が生じる。つまりCは（ということはAもBも)，独立性が高くないということである。

以上のことを例を挙げて考えてみよう。

今，1秒，2秒，3秒という3種類のタ

イマを必要とするプログラムがあったとする。この部分をモジュールとみなして，サブルーチンにする。

図6がその例である。このプログラムは，1，2，3秒の間隔をおいてBEEP音を発生するものである。

モジュール4（md4）の使い方は，ある雑誌に掲載されていたものを，そのまま流用した。このリスト中，md1, md2, md3は特に問題はない。しかしmd4は，モジュールとして不都合である。

この例では260行を1度実行すると，ほぼ1秒の時間かせぎをすることになっている。したがって，たとえば2秒のタイマとして使うなら，250行の変数TMを2にセットしなければならない。ここではそのTMへの値セットのしかたを，高度な（？）テクニックで実現している。

つまり，210行から入るとTMは1になり，230行からならTM＝2，240行ならTM＝3になる，という仕掛けなのだ。でも，うまい！と思ってはいけない。

これはモジュールの定義③「1つの入口と1つの出口を持つ」に違反している。何も，モジュールの例題として考えなくとも，こういうプログラミングは間違いである。

次に入口が1つであることにこだわると，プログラムは図7のようになる。この例では，モジュール自体を1秒用，2秒用，3秒用と完全に分けた。入口は1つ，出口も1つ，先述の不都合はたしかになくなった。

しかし，今，次のような変更を考えてみよう。このタイマが，1，2，3秒用ではなく，実はおのおの10, 20, 30秒用だったという場合である。それには，カウンタJの部分を，

```
FOR J=1 TO 10000:NEXT
```

と変える必要があるが，図7の場合だと，md4-1, md4-2, md4-3の3つのモジュールすべてに，変更の影響がおよぶ。

つまり，たったひとつの機能を変更するために，実際には3つのモジュールについて変更作業を行わなければならないわけで，これは，md4-1, md4-2, md4-3の各モジュールの独立性が低いことを示している。

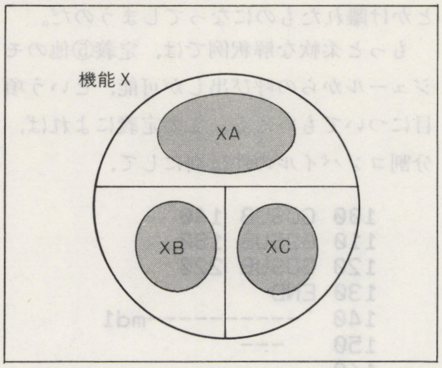
このケースでは，たとえば図8のようなモジュール化の方が適切である。このリストでmd4は完全にクローズしている。

あまり上手なプログラム例ではなかったが，モジュール化というものについて，多少はわかっていただけたかと思う。

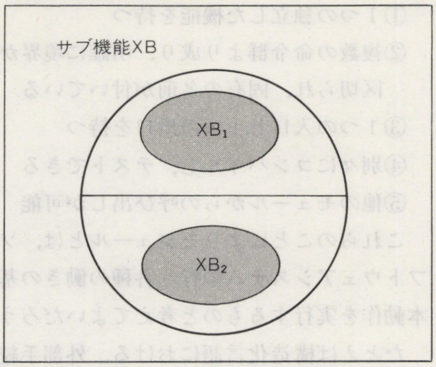
最後に，構造化設計を行う上での心得をいくつか述べよう。

①トップダウンが万能とは考えないこと
②明確なステップを意識しながら設計をすすめること
③各ステップでは，もっとも容易な，最小限の決定のみを行うこと
④単純な設計にするよう努力すること
⑤機能は理解できる大きさにまで分割して，モジュール化すること
⑥入力ミス，エラー検出，回復，保守，拡張についても設計で考慮しておくこと
⑦構造図，HIPO，擬似コードなどを有効に利用して，設計結果を文書化する
⑧関係部門の承認を得てからコーディングにはいること

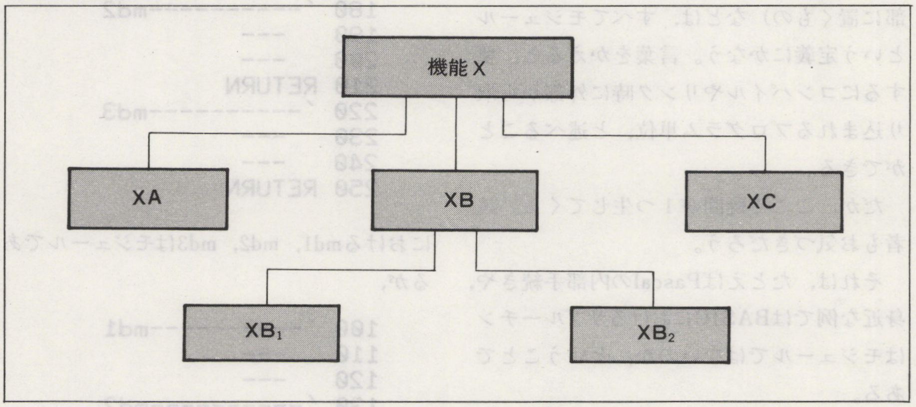
▼図2　機能分解1



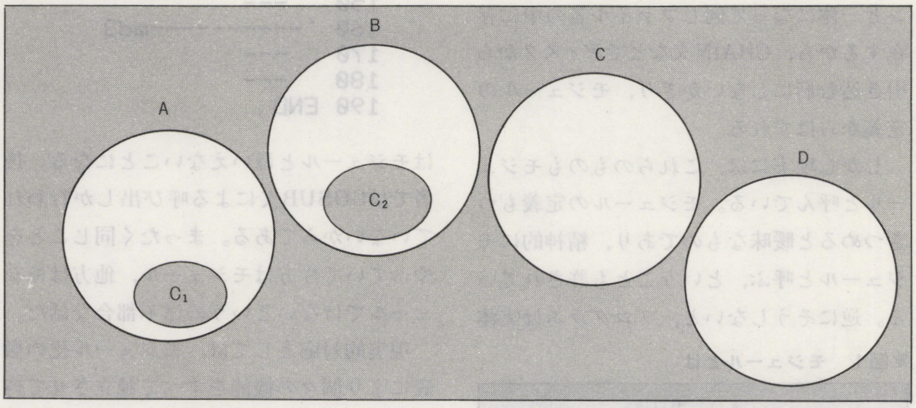
▼図3　機能分解2



▼図4　階層化



▼図5　機能の独立性



▼図6

```
100 '---------- md1
110 GOSUB 210
120 BEEP
130 '---------- md2
140 GOSUB 230
150 BEEP
160 '---------- md3
170 GOSUB 240
180 BEEP
190 END
200 '
210 '---------- md4
220 TM=1:GOTO 250
230 TM=2:GOTO 250
240 TM=3
250 FOR I=1 TO TM
260     FOR J=1 TO 1000:NEXT
270 NEXT
280 RETURN
```

▼図7

```
100 '---------- md1
110 GOSUB 210
120 BEEP
130 '---------- md2
140 GOSUB 260
150 BEEP
160 '---------- md3
170 GOSUB 310
180 BEEP
190 END
200 '
210 '---------- md4-1
220 FOR I=1 TO 1
230     FOR J=1 TO 1000:NEXT
240 NEXT
250 RETURN
260 '---------- md4-2
270 FOR I=1 TO 2
280     FOR J=1 TO 1000:NEXT
290 NEXT
300 RETURN
310 '---------- md4-3
320 FOR I=1 TO 3
330     FOR J=1 TO 1000:NEXT
340 NEXT
350 RETURN
```

▼図8

```
100 '---------- md1
110 TM=1:GOSUB 210
120 BEEP
130 '---------- md2
140 TM=2:GOSUB 210
150 BEEP
160 '---------- md3
170 TM=3:GOSUB 210
180 BEEP
190 END
200 '
210 '---------- md4
220 FOR I=1 TO TM
230     FOR J=1 TO 1000:NEXT
240 NEXT
250 RETURN
```

モジュールの論理設計

モジュールの論理設計には，概要フローに相当する大まかな処理手順設計と，実際のコーディング作業がある。ここでは，コーディング作業に比重をおいて話をすすめる。

論理の構造化については，有名なストラクチャード定理というものがある。それは，

――プログラムが1つの入口と1つの出口をもつように設計されていれば，基本的な3つの構造単位の組み合せによって，どんな論理でも記述できる。3つの構造単位とは，

順次（SEQUENCE）
選択（IF～THEN～ELSE）
繰り返し（DOWHILE）

である。

これがいわゆる基本3構造というものである。

実際のプログラミングにおいては，これ以外にもさまざまなテクニックがあり，それらは経験的にしかマスタできない傾向がある。したがって，基本3構造さえ知れば，その日からすぐに優秀なコーディングができる保証はまったくないが，この3つがたしかに基本中の基本であることは，間違いない。

ところで，プログラムは本来，ある特性をもっている。それは，

――上から下へ流れる

ということである。だから，この基本特性を忠実に守っていれば，虫のはいり込む余地は限定されることになるわけだ。たとえば，

```
100 INPUT A
110 INPUT B
120 ADD=A+B
130 PRINT "add=";ADD
140 SUB=A-B
150 PRINT "sub=";SUB
160 END
```

といった，単純に上から下へ流れるプログラムの場合には，たぶん正常に動くだろうことは容易に想像がつく。もし虫があったとしても，それはまったくの単純なミスであるケースが多い。

ところがこのプログラムが，あるルーチンの一部だったとして

```
230 ---
240 INPUT A
250 INPUT B
260 GOTO 150
270 ADD=A+B
280 PRINT "add=";ADD
290 SUB=A-B
300 PRINT "sub=";SUB
310 ---
```

のように書かれていたらどうだろうか。おそらく260行のGOTO 150の部分にて，

――ちょっと待てよ

と，かまえてしまうはずである。

頑丈なはずの堤防が，針の穴から決壊をはじめるように，このプログラムでも何かあるとしたら，それは，GOTO 150が糸口になるのである。

順次の構造は図9のようになる。ここでSI……は文（statement）を意味している。

順次は，言語によっては，複合文と呼ば

れたりする。たいていの構造化言語では，beginおよびendにて，複合文の範囲を指定する。

順次構造は単純であるだけに，もっとも安定したプログラムを作成できる。

しかし，通常，それだけでは複雑な論理を表現するには限度がある。プログラム設計の中に「もし」という論理が1つでも出てきたら，それはもう順次だけでは表現できない。

その場合には，選択文が採用されることになる。

ところがアセンブラにおいては，選択といっても条件つきジャンプ命令でどこかへ飛ぶだけなので，飛び先を目で追いかけなければならず，誤りの原因になることが多い。これはBASICの場合でも，GOTO文の乱用で起こりうることである。

構造化言語における選択文は，IF～THEN～ELSE文ともいわれ，図10のように表現される。

このうち，

IF A THEN S1

は，

IF A THEN S1 ELSE S2

の文のうち，ELSE以下が空文（何もしない文）になったものと考えられるから，両者は同一のものである。

IF～THEN～ELSE文で重要なことは，どこで分岐が始まり，どこで再びルートが結合するかがはっきりしていることである。字下げなどを用いてうまく配置すると，その関係が視覚的に判断できることになる。

ここからここまでは選択文であるということが，すぐに理解できるなら，それは理解度という点では，順次構造がたまたま2本に変形しているだけで，全体構造は順次構造と何も変わりないのである。

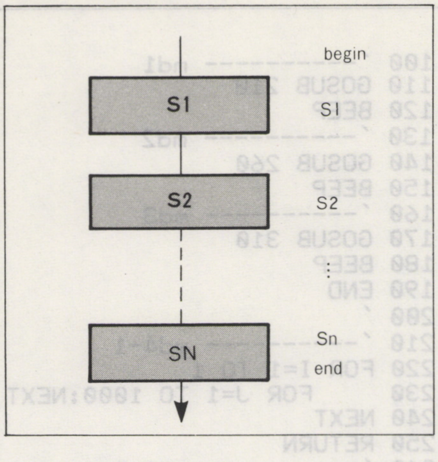
構造化言語の各種構文は，いろいろな論理構造を，いかにして基本である順次構造に近づけるかという努力の結果ととらえてもよいであろう。

IF A THEN S1 ELSE S2

のELSE以下が，数を増すと，それは図11のようなCASE文といわれる形になる。

CASE文はIF～THEN～ELSE文の拡張

▼図9　順次構造



と考えることができる。BASICにおいては，

ON N GOTO

などという文がこれに近い。

ただし，ON N GOTO文においては，単にどこかの行番号へ多重ジャンプするだけであり，これではプログラムの明解度はあまり高くならない。図11の例のS1～S4においては直接，実行文が記述される。それだけストレートな表現となるわけだ。

基本3構造の最後は，繰り返し文である。これは図12のようになる。

繰り返し文には，

FOR～NEXT文

WHILE文

REPEAT文

などのタイプがあり，使用法は少しずつ異なっている。その違いについて，ここでは述べないが，ここでも大事なことは繰り返し文の採用により，GOTO文がなくなるということである。

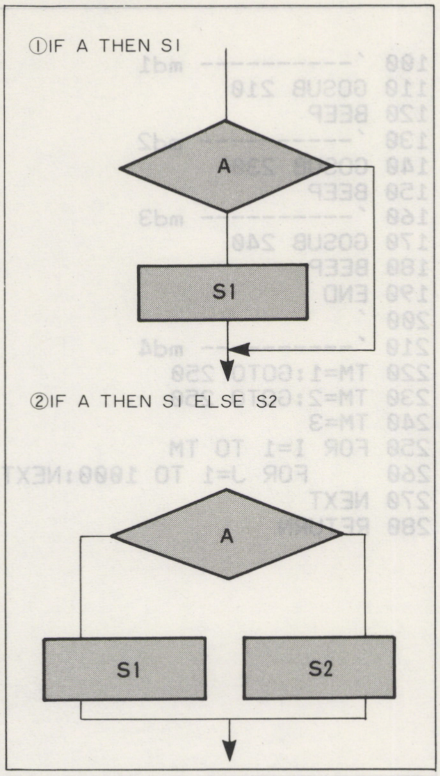
GOTO文には，不利な点が2つある。

1つは，どこへ飛ぶかということが不明な点，もう1つは何のために飛ぶのかが不明な点である。

繰り返し文では，NEXTなどの終端文により，どこへ飛ぶかをはっきりさせ，またFORやWHILEというキーワードにより，何のために飛ぶのかを積極的に明示する効果をもっている。曖昧なGOTO文としての機能を，構文の中に隠してしまっているわけだ。

標準的なプログラミングレベルに達した

▼図10　選択構造



人なら，たしかにこの3つの構造単位を用いて，どんな論理も記述できる。もし記述できない場面に遭遇したら，それは設計自体を一度疑ってみる価値が充分にあるだろう。

ただし，基本3構造だけで記述できるといっても，それらが素直な形で利用できるということではない。場合によっては，3つの構造が複雑に入りくんで，ひとつの表現となる場合もありうる。

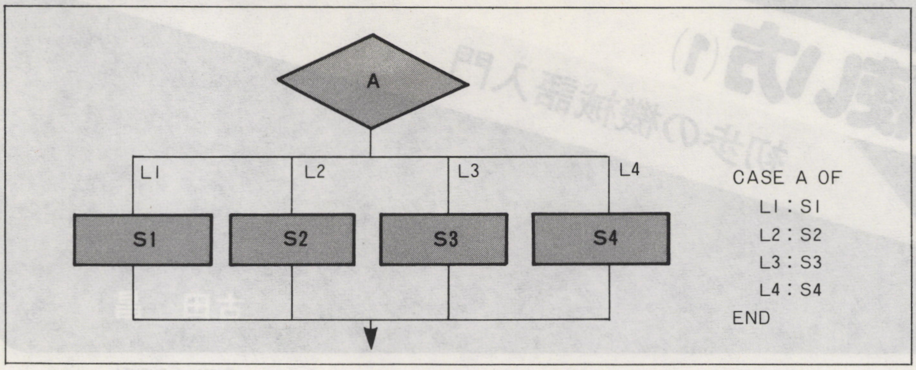
たとえば，図13のような例となる。

ここではBASIC言語にこだわらずに，プログラミング行為そのものを対象にして，理論的な話をすすめたが，BASICを対象にすると，あらゆる面で構造化はやりづらくなる。

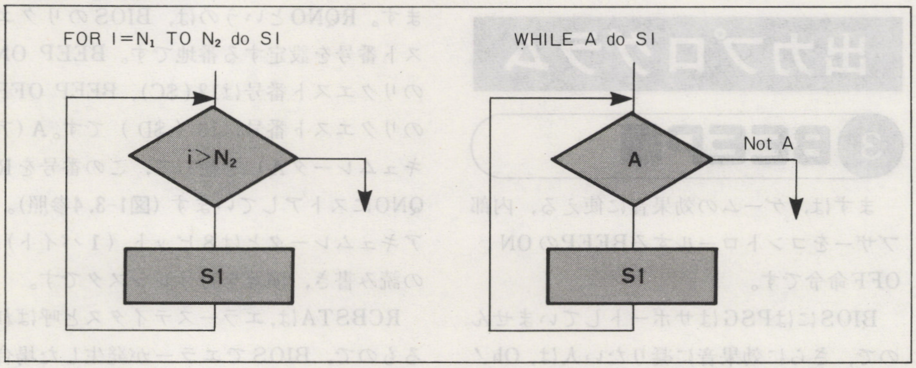
それは非構造化言語BASICの限界ともいえるだろう。

たとえば，GOTO文を追放すべき，と主張しても，BASICでGOTO文をゼロにすることはできない。しかし，だからといって，これをあきらめることはない。何らかの自己規制でGOTO文の使用を最小限にする工夫はあってもいいだろう。それには，

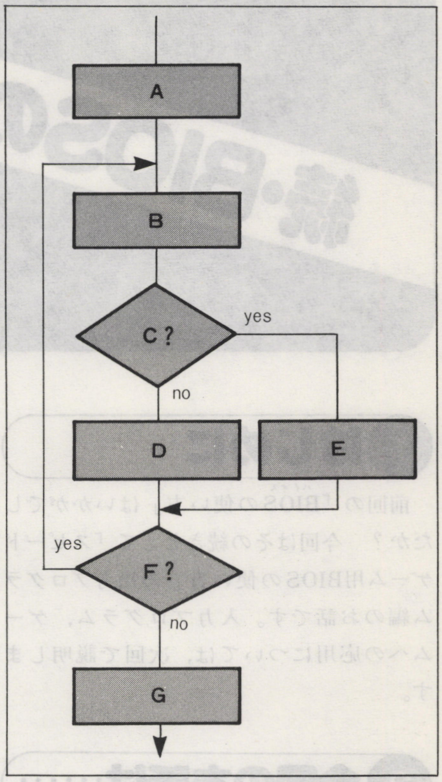
▼図11　選択構造（多重分岐）



▼図12　繰り返し文



▼図13　基本３構造の組み合せ



—単なるGOTO文は使用禁止にするが，構造化表現をするためなら，やむをえない

といったことになる。

このような方針を決定した場合，残されたGOTO文については，本当にそれがやむなく使用されているのかどうかは，厳重な監視が必要となろう。

たとえばGOTO文のルートがからみあっていたり，また若い番号の方へ飛んでいるGOTO文の場合には，再検討の余地がある。

最　近　の　動　き

構造化プログラミングは，漠然としたわかりにくいものである。使われるようになってから，すでに10年以上たっているのに，まだ理論的完成をみないし，予想されている利点を使いきってもいない。

構造化プログラミングの技術は，現在も進歩していることは事実だが，まだまだやるべきことが多くあるのも事実である。

現在進められているプロジェクトの中で，注目すべきものに，Carnegie-Mellon 大学のGandalfシステムというのがある。

これはソフトウェアの開発環境を効率的にしようというものである。

・システムバージョン制御
・プログラムの構築
・プロジェクト管理

などを目的としたもので，操作としては，

・ソフトウェアの新規作成
・ソフトウェアの修正・変更・管理
・プログラムの実行とデバッグ
・ドキュメントの保守・管理

などを行うことができる。

そのソフトウェアシステムの記述方法をみると，以下のように４つのセクションより構成されている。

①box
　moduleまたはbox自身の集合
②module
　機能リストつきversionの集合
③version
　作成日付順revisionの集合
④revision
　機能の実現をする実際のプログラム

この他にもALOE（A Language Oriented Editor）と呼ばれる構文型エディタの採用など，従来の構造化プログラミング手法を基盤にした斬新なアイディアがもりこまれている。

これらの新しい成果を知るにつけ，なるほど構造化プログラミングの奥は深いと，また思い知らされる。

for FM-7,8,11

続・BIOSの使い方(1)

初歩の機械語入門

古田　昌

©1983 A. FURUTA

1 はじめに

前回の「BIOS(バイオス)の使い方」はいかがでしたか？　今回はその続きとして「スピードゲーム用BIOSの使い方」の出力プログラム編のお話です。入力プログラム，ゲームへの応用については，次回で説明します。

2 今回のお話は……

これはVol.6の「BIOSの使い方」の続きです。それを読んでいない人は，まずそちらを読んでください。

前回で1ヵ所誤りがありました。47ページ右段10行目「……IF A＝＄30……」は，「……IF A<>＄30……」でした。訂正し，おわびいたします。

さて前回はBIOSの「INPUT」と「OUTPUT」のお話をしました。BIOSには，ほかにもいろいろな入出力があります。今回はその中から，高速ゲームに便利な命令を選んで紹介します。

なぜ高速ゲーム用のものかというと，ビギナーがマシン語に興味をもつのは，高速ゲームができるから，という理由がいちばん大きいと思うからです（かくいう私もそうでした）。しかし，FMシリーズでは入出力がよくわからないために，マシン語に対して興味を失う場合が多いのです。そんなことのないように，高速ゲーム用のBIOSの使い方を詳しくお教えしましょう，というのが今回のテーマです。

出力プログラム

3 BEEP編

まずは，ゲームの効果音に使える，内部ブザーをコントロールするBEEPのON，OFF命令です。

BIOSにはPSGはサポートしていませんので，さらに効果音に凝りたい人は，Oh！FM Vol.4のP.89「PSGサウンドエディタ」をご利用ください。

図1-1がダンプリスト，図1-2がソースリストです。まずダンプリストを打ち込んで，セーブしてください。EXEC&H4000で実行すると，少しの間「ピーッ」とブザーが鳴り続けてから，Ready状態になります。

うまく鳴らないか，鳴りっぱなしの人は，ストップリセット（またはリセット，ロード），デバック，セーブを繰り返して，バグがなくなるまで続けましょう。

プログラムの解説にいきます。

＄4000～＄400BがBEEP ONコールルーチン，＄400C～＄4012がタイマルーチン，＄4013～＄401EがBEEP OFFコールルーチン，＄4020，＄4021がRCB領域です。

BEEP ONをコールすると，ブザーが鳴り出しますから，タイマルーチンで間をとって，BEEP OFFでブザーを切ります。

BEEPルーチンでは，RCB（Request Control Block，詳しくは前回を参照）領域は2バイトで済みます。RQNOというのは，BIOSのリクエスト番号を設定する番地です。BEEP ONのリクエスト番号は12（＄C），BEEP OFFのリクエスト番号は13（＄D）です。A（アキュムレータA）を使って，この番号をRQNOにストアしています（図1-3，4参照）。アキュムレータとは8ビット（1バイト）の読み書き，演算を行うレジスタです。

RCBSTAは，エラーステイタスと呼ばれるもので，BIOSでエラーが発生した場合は，キャリーが立つとともに，ここにエラー番号がセットされます（エラー番号は，マニュアル参照）。BEEPでは，エラーは発生しないので，常に正常値（＝＄0）を示します。

「LDX ＃＄4020」により，Xレジスタに RCBのアドレスが設定されていますが，「RCB EQU ＊」によって，RCBは現在のアドレスを示します。このためRCBとRQNOは，同じアドレスを持ちます。

最後にタイマルーチンですが，このタイマルーチンは，Xレジスタを使っています。「LDX ＃0」がCLRX，「LEAX －1，X」がDECXと同じ意味をもちます。「BNE LBL1」でゼロフラグが立っていないときは，LBL1にブランチして，「LEAX」を繰り返します。

ここで注意してもらいたいのですが，ゼ

図1-1

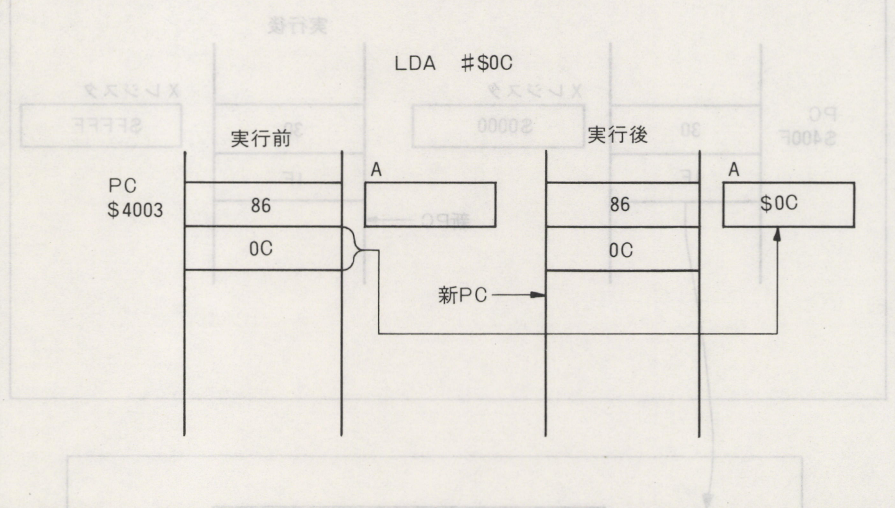
```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
4000  8E 40 20 86 0C B7 40 20 :97
4008  AD 9F FB FA 8E 00 00 30 :FF
4010  1F 26 FC 8E 40 20 86 0D :C2
4018  B7 40 20 AD 9F FB FA 39 :91
4020  0C 00 00 00 00 00 00 00 :0C
```

図1-2

```
                                 NAM     BEEP
                                 TTL     ** BIOS BEEP **
         4000                    ORG     $4000
                         *
               FBFA      BIOS    EQU     $FBFA
                         *
               4000      ENTRY   EQU     *
                         *
         4000 8E  4020   BEEPON  LDX     #RCB
         4003 86  0C             LDA     #12
         4005 B7  4020           STA     RQNO
         4008 AD  9F FBFA        JSR     [BIOS]
                         *
         400C 8E  0000           LDX     #0
         400F 30  1F     LBL1    LEAX    -1,X
         4011 26  FC  400F       BNE     LBL1
                         *
         4013 8E  4020   BEEPOF  LDX     #RCB
         4016 86  0D             LDA     #13
         4018 B7  4020           STA     RQNO
         401B AD  9F FBFA        JSR     [BIOS]
                         *
         401F 39                 RTS
                         *
               4020      RCB     EQU     *
         4020  0C        RQNO    FCB     12
         4021  00        RCBSTA  FCB     0
                         *
               4000              END     ENTRY
TOTAL ERRORS 00000--00000
TOTAL WARNINGS 00000--00000

PROGRAM BEGIN ADDR=4000
PROGRAM END   ADDR=4021
PROGRAM ENTRY ADDR=4000
```

図1-3



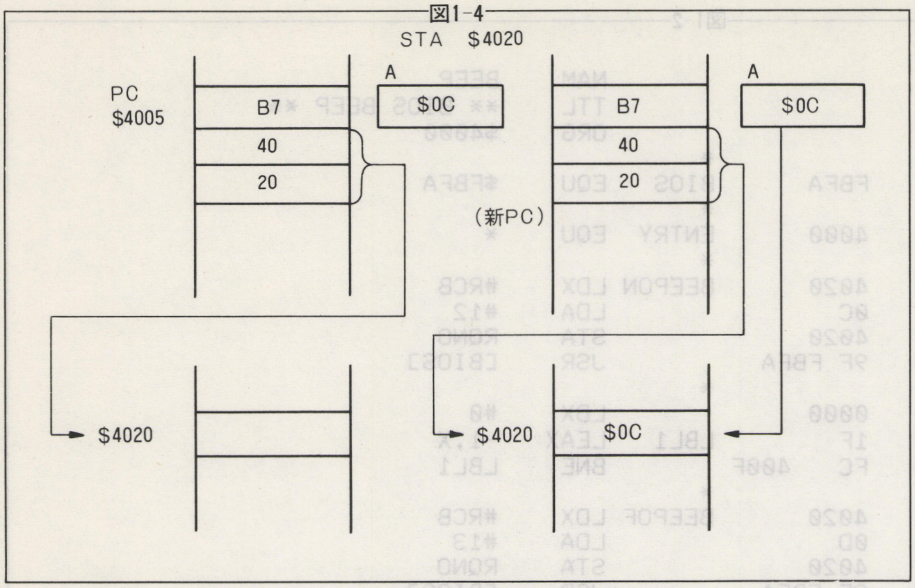
ロフラグというのは，コンディションコードレジスタ（前回参照）の２ビット目のフラグで，その直前に実行した演算結果が０になった場合に立つフラグです。LEA命令は演算ではないため，通常はフラグは変化しませんが，X，Yレジスタの場合のゼロフラグに限り，フラグが変化します。このため，X，Yレジスタもタイマやカウンタとして使えます。

このLEA命令というのは，X，Yレジスタおよび U，S スタックポインタに，指定アドレスそのものの値を（アドレスの内容ではなしに）そのままセットする命令です。たとえば，「LEAX〔3000〕」は「LDX 3000」に同じです。

「−1，X」というアドレスは，Xレジスタの値から，１引いたアドレスを意味します。このアドレスを，インデックストアドレスといいます。（図1-5参照）。Xの他にY，S，U，PCが使えます。LD（ロード）では，このインデックスアドレスの示すアドレスが実効アドレスになりますが，LEA命令では，インデックストアドレスをそのまま読み込むので，レジスタを中心とした相対値を読み込めるという利点があります。

ここで，少しループの話をしておきます。このタイマルーチンのように，同じところをぐるぐる回るルーチン（命令過程）を「ループ」といいます。F-BASICの「FOR～NEXT」ループ，「WHILE～WEND」ループなどは，おなじみのことと思います。

マシン語では，ブランチ命令を使って，ループをつくります。BASICと違って，ループ内への出入りはまったくの自由ですが，ループの終了がうまくいかないと，いつまでもループの中を堂々巡りしてしまうこと



があります。これを「無限ループにおちこんだ」といいます。

マシン語でのミスは，この無限ループにおちこむ場合が一番やっかいなもので，そのくせしょっちゅうおきるのです。対策としては，ループの終了のチェックに用いているレジスタを，他には絶対使用しないか，どこかにループ制御データを退避させておくしかありません。

「LDX ＃0」の0の値を変えると，鳴っている長さが変わります。違った長さをいくつか組み合せると，いろいろな音がでますが，今はこの辺で説明を終わります。

4 SUBOUT編

お待たせしました，やっとBIOS-SUBOUTの番がまわってきました。FMシリーズでは，メインとサブの2つのCPUを持っていますが，サブCPUにコマンドを転送・実行するのが，このSUBOUTです。これを使えば，サブCPUを動かすことができます。

サブシステムを動かすにはHLT（ホールト），MOVE（ムーブ）などややこしい手続きが必要ですが，このSUBOUTを使えば，そういうことは一切BIOSがやってくれます。

実は以前に，Vol.3「サブシステムプログラム転送・実行」(FILES Oh!FM p.232) でも紹介されていますが，今回はもう少し説明をつけ加えて紹介しましょう。

図1-6がダンプリスト，**図1-7**がソースリストです。打ち込み，セーブの後，実行すると，画面の背景色が青になります。つまり，BASICの「COLOR,1：CLS」と同じ作業をしています。

うまくいかなかった人は「リセット，ロード，デバッグ，セーブ，実行」を，うまくいくまで繰り返しましょう。

解説に移りますので，ソースプログラム(**図1-7**) をみてください。

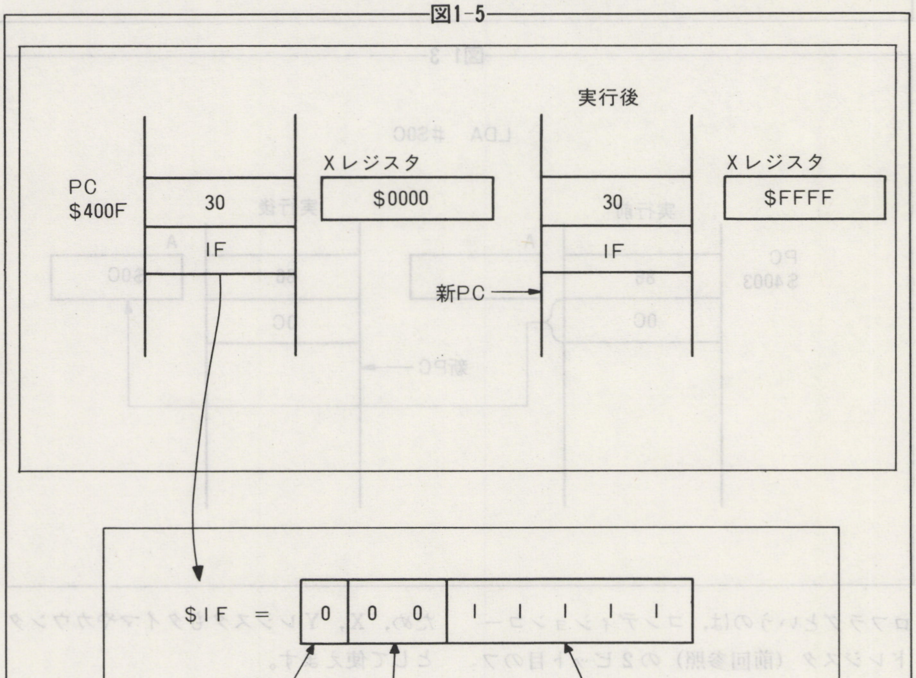
まずRCBのパラメータを定義します。

SUBOUTのリクエスト番号16($10)をRQNOに定義して，RCBSTA領域を確保したら，データバッファのアドレス「DATA」をRCBDBA(RCB Data Buffer Address)に定義して，転送バイト長(1～128) を，RCBLNH (RCB Long High) に定義します。

RCBSTAは，BIOSのシステムエラーがあった場合に，エラーコードが入る領域です。RCBのパラメータの設定さえきちんとしていればエラーは起こりませんので，ここでは常に正常値(0)になります。

RCBDBAには，サブシステムに転送するデータ領域の先頭番地をセットします。この番地から，RCBLNHに定義されているバイト数だけ，そのまま中間領域に転送されます。

中間領域について説明しておきましょう。**図1-8**をみてください。中間領域というのは，メインとサブシステムの間のデータの受け渡しが行われるアドレス領域のことで，128バイト ($80) の長さがあります。メイン側では，＄FC80～＄FCFFのアドレスを持ち，サブ側では＄D380～＄D3FFのアドレスを持っています。メイン側でサブをコマンド待ち状態 (HLT（ホールト）) にすると，メインからこの領域への書き込みが可能になり，データの書き込み後にメイン側がサブCPUを



実行状態(MOVE(ムーブ))にすると，サブ側で中間領域の読み出しが可能になって，このときのデータにしたがって，コマンドを実行します。

SUBOUTでは，サブCPUのHLT，データ領域のデータを中間領域への書き込み，サブシステムのMOVEを自動的にやってくれます。

RCBLNHに中間領域に転送するデータのバイト数を設定しますが，中間領域が128バイトしかないので，これを超えてはいけません。この値は常に128にしておいても，とくにさしつかえありません。

データ転送後は，サブCPUの終了を待たずに，復帰します。このため，サブシステムに生じるエラーのチェックは行いません。(ただしFM-11に関しては，サブシステムへのコマンドコードが0かどうかがチェックされます)。

次にサブシステムへの転送データ列につ

図1-6

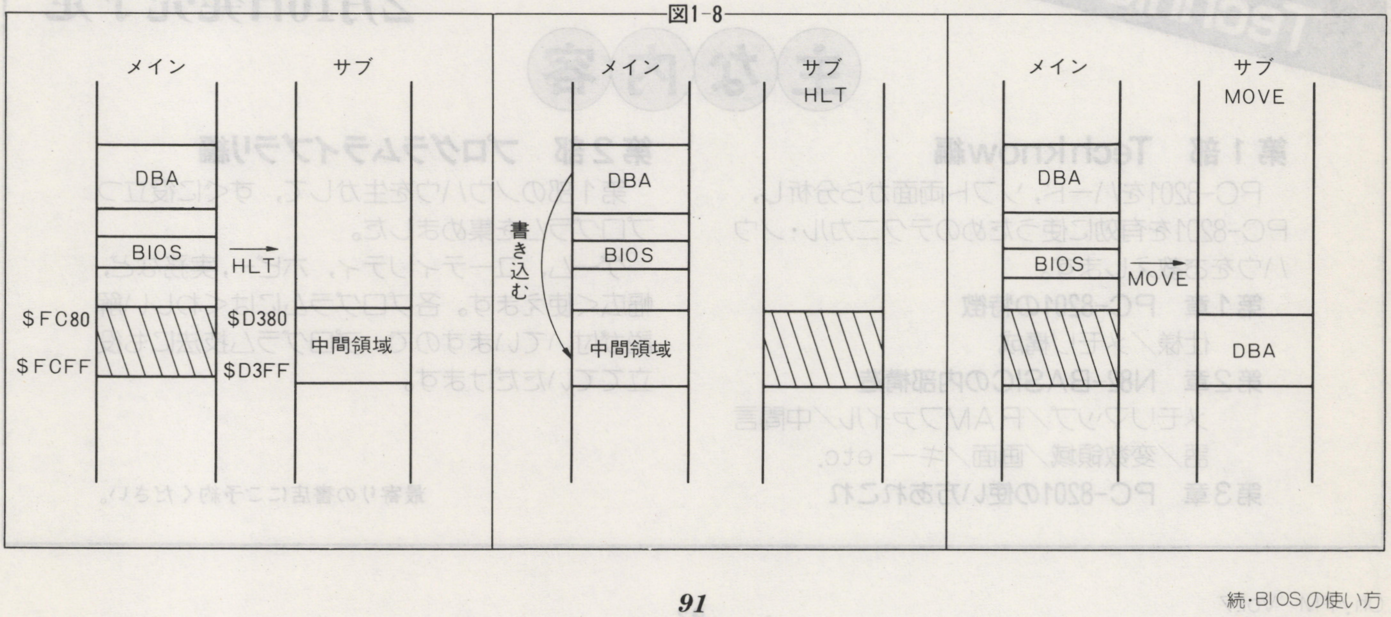
Add	+0	+1	+2	+3	+4	+5	+6	+7	Sum
4000	8E	40	08	AD	9F	FB	FA	39	:50
4008	10	00	40	0E	00	80	00	00	:DE
4010	0D	00	01	07	00	00	00	00	:15

図1-7

```
                                NAM     SUBOUT
                                TTL     ** BIOS SUBOUT **
        4000                    ORG     $4000
                        *
                FBFA    BIOS    EQU     $FBFA
                        *
                4000    ENTRY   EQU     *
                        *
        4000 8E 4008    START   LDX     #RCB
        4003 AD 9F FBFA         JSR     [BIOS]
        4007 39                 RTS
                        *
                4008    RCB     EQU     *
        4008    10      RQNO    FCB     16
        4009    00      RCBSTA  FCB     0
        400A    400E    RCBDBA  FDB     DATA
        400C    0080    RCBLNH  FDB     128
                        *
                400E    DATA    EQU     *
        400E    00      E       FCB     0
        400F    00      CN      FCB     0
        4010    0D      C       FCB     $0D
        4011    00      W       FCB     $00
        4012    01      BC      FCB     $01
        4013    07      FC      FCB     $07
                        *
                4000            END     ENTRY
TOTAL ERRORS 00000--00000
TOTAL WARNINGS 00000--00000

PROGRAM BEGIN ADDR=4000
PROGRAM END   ADDR=4013
PROGRAM ENTRY ADDR=4000
```

図1-8



いて説明します。図1-7をみてください。

E (Error) には，サブシステムへのコマンドやパラメータに誤りがあった場合に設定されます。正常に処理されたときの値は0です。

CN (Continue, 7/8はSTATUSとも書く) は，Sとも書き，データ列が128バイトを超えたときに立てる継続フラグが入っています。

EとCNは普通はどちらも0にしておけばいいですから，今回は詳細を略します。

C (Command) がサブシステムへのコマンドコードです。「$0D」は「ERASE 2」のコマンドコードです。コマンドコードとパラメータを図1-9に示します。

W以下は「ERASE 2」に属するパラメータで，W＝0は全画面消去 (CLS0)，BC＝1は背景色青，FC＝7 (FM-11ではAT＝7) は文字色白であることを指定しています。色を変えて確かめてください。

最後に使い方をまとめておきます。

RCB領域にリクエスト№16 ($10)，エラーステイタスバッファ，データ領域番地，転送バイト数128 ($80) をセットする。次にデータ領域に「0，0，コマンドコード，コマンドに対応するデータ列」をセットし，XレジスタにRCBのアドレスをロードして，BIOSをコールする。

図1-9 コマンド＆パラメータ

相対値	記号	内容	値
0，1			Don't care
2	C	コマンドコード	$0D
3	W	範囲	0＝全画面 1＝スクロールモード画面 2＝ページモード1画面 3＝ページモード2画面
4	BC	背景色	0～7 (ただしW＝0のとき有効)
5	FC	文字色	0～15

お待ちかね！
PC-8201読本ついに登場！
特徴，活用そしてプログラムライブラリ！
8201ユーザー必読の書!!

Tech Know Lib 8200
Technical know-how and program Library

B5判 256ページ
予価 1,800円
システムソフト編著
日本ソフトバンク発行
2月10日発売予定

主な内容

第1部 Techknow編
PC-8201をハード，ソフト両面から分析し，PC-8201を有効に使うためのテクニカル・ノウハウをお教えします。
第1章 PC-8201の特徴
仕様／メモリ構成
第2章 N82-BASICの内部構造
メモリマップ／RAMファイル／中間言語／変数領域／画面／キー etc.
第3章 PC-8201の使い方あれこれ

第2部 プログラムライブラリ編
第1部のノウハウを生かして，すぐに役立つプログラムを集めました。
ゲーム，ユーティリティ，ホビー，実務など，幅広く使えます。各プログラムにはくわしい解説が付いていますので，プログラム技法にも役立てていただけます。

最寄りの書店にご予約ください。

MSXマシン FM-X 続報

本GO 誠司

本誌Vol.6で速報したように，FMシリーズに新たにFM-Xが加わった。今号では，もう少し深く，MSXについて，またFM-7との接続時について，わかったところを報告してみよう。

MSXマシンの仕様

MSXは，パソコンの規格共通化（ハードやソフト）のために，去年6月にマイクロソフトと(株)アスキーより提案されたものであり，FM-Xは，このMSX規格のパソコンである。'83年12月1日現在，MSX規格のパソコンは，他にヤマハ，ソニー，東芝，三菱，日立，松下，三洋，ビクターから発表されており，各社それぞれなんらかの独自性を持たせている。FM-Xにおいては，既報のように，「FM-7と接続できる」ことが最大の特色であるが，まずはFM-Xの基本であるMSXについて，少し述べてみよう。

☆MSXのハードウエア

MSXのハードウェア構成は，以下のとおりである。

①CPU：Z-80Aソフトコンパチブル
　クロック3.579545MHz
②メモリ：ROM32KB(MSX BASIC)
　RAM 8KB以上
③VDP（CRTC）：TMS9918Aコンパチブル
④PSG：AY-3-8910コンパチブル
⑤PPI：i8255コンパチブル
⑥画面表示：テキスト最大40文字×24行，グラフィック256×192ドット，カラー16色
⑦CMT：FSK方式，1200/2400ボー
⑧サウンド：8オクターブ，3重和音出力
⑨キーボード：英数，ひらがな，カタカナ，グラフィック記号対応，JIS配列/五十音順配列対応
⑩ROMカートリッジ：ROMカートリッジ，拡張バスに対応するスロットを持つ
⑪I/Oバス：カートリッジバスは50ピン，カードエッジ型
⑫プリンタ：8ビットパラレルインタフェース。最小システムでは付属しない場合もある
⑬ジョイスティック：1または2本接続可
⑭フロッピディスク：各社対応
⑮漢字機能：各社対応

以上がMSXのハード構成である。①のCPUにZ-80が選ばれたのは，やはり，ソフト的ノウハウの蓄積からであり，6809ファンとしては，情けない気がする。②のROMおよびRAMは，ともに拡張可能となっており，標準FM-XではROM32Kバイト，RAM16Kバイトが装備されている。またのちに述べるFM-7と接続することにより，ROM48Kバイト，RAM32Kバイトとなる。③は，TVゲームなどでよく使われているLSIで，32枚のスプライト画面，16色のカラー表示，256×192ドットのグラフィックなどが可能な画面表示LSIである。FM-Xでは，その相当品であるTMS9928Aが使われている。④は，FM-7の音楽用LSIと同じものである。⑤は，80系のインタフェースとして有名すぎるくらい有名なものである。⑨のキーボードは，ひらがな・カタカナをサポートしているため，**表1**のようにキャラクタコードは，かなり，変則的になっている。また，キー配列はJISと五十音順があるが，FM-XではJIS配列，ひらがな表示となっている。⑫ のプリンタは，FM-Xのみ使用時は，プリンタI/F・RAMカートリッジ(オプション)が必要で

あり，また，FM-7と接続すればFM-7側のプリンタが使える。⑬のジョイスティックは，④のAY-3-8910の汎用ポートを使用し，入力4ビット，出力1ビット，双方向（入出力）2ビットとなっている。正論理TTLレベルで，2ポートまでサポートされるため，ジョイスティック2本まで接続可能となっている。

表1　キャラクタコード表

下位4ビット＼上位4ビット	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		π		0	@	P	′	p	♠			ー	タ	ミ	た	み
1	月	┴	!	1	A	Q	a	q	♥	あ	。	ア	チ	ム	ち	む
2	火	┬	”	2	B	R	b	r	♣	い	「	イ	ツ	メ	つ	め
3	水	┤	#	3	C	S	c	s	◆	う	」	ウ	テ	モ	て	も
4	木	├	$	4	D	T	d	t	○	え	、	エ	ト	ヤ	と	や
5	金	┼	%	5	E	U	e	u	●	お	・	オ	ナ	ユ	な	ゆ
6	土	│	&	6	F	V	f	v	を	か	ヲ	カ	ニ	ヨ	に	よ
7	日	─	’	7	G	W	g	w	ぁ	き	ァ	キ	ヌ	ラ	ぬ	ら
8	年	┌	(	8	H	X	h	x	ぃ	く	ィ	ク	ネ	リ	ね	り
9	円	┐	)	9	I	Y	i	y	ぅ	け	ゥ	ケ	ノ	ル	の	る
A	時	└	*	:	J	Z	j	z	ぇ	こ	ェ	コ	ハ	レ	は	れ
B	分	┘	+	;	K	〔	k	{	ぉ	さ	ォ	サ	ヒ	ロ	ひ	ろ
C	秒	╳	,	<	L	¥	l	¦	ゃ	し	ャ	シ	フ	ワ	ふ	わ
D	百	大	−	=	M	〕	m	}	ゅ	す	ュ	ス	ヘ	ン	へ	ん
E	千	中	・	>	N	∨	n	～	ょ	せ	ョ	セ	ホ	゛	ほ	
F	万	小	／	?	O	ー	o		っ	そ	ッ	ソ	マ	゜	ま	

☆MSXのソフトウェア

MSXのソフトウェアの主たるもの，すなわちBASICには，標準で14桁10進の倍精度演算および関数計算可能なVer 4．5のビジネス仕様マイクロソフト拡張BASIC（32Kバイト）が使用されている。演算は整数演算を除いてBCD（2進化10進）演算で行われ，バイナリ演算のような誤差を生じないのが大きな特徴である。もちろん整数型も用意され，−32768～＋32767までの整数を高速に扱うことができる。また，2進数を直接扱えることも特徴となっている。

MSXでは互換性と拡張性が重視されているので，BASICの先頭部分には，CP/MやF-BASICのBIOSと同じ様なBIOSが置かれている。これはBASIC内部の機械語エントリアドレスをまとめたもので，これを利用することにより，機械語プログラムの開発が簡易になり，またプログラムの互換性を保つことが可能になる。これらの内容は，ユーザーはもちろんのこと，ソフトウェアメーカーにも公開されるため，豊富なソフトウェアの流通を可能とすることだろう。

MSX BASICのコマンド，ステートメント，関数などは，F-BASICがマイクロソフト系BASICなので，FMユーザーはほとんど抵抗なく入っていける。

FM-X＋FM-7

FM-Xは，以上のようなMSXマシンであるが，FM-7と接続することによって，他社のMSXマシンにない特色が出せる。

7との接続には，「FM-7インタフェース」（オプション）を使う。インタフェースケーブルの片方には，FM-7インタフェースカードがついており，これをFM-7本体の上部拡張スロット右端（Z-80カードを差し込むところ）に接続し，もう片方のFM-Xインタフェースカートリッジを，FM-X本体の60ピン拡張スロットに接続する。

FM-Xインタフェースカートリッジ内のROMには，拡張MSX BASICのシステムが入っている。

FM-7インタフェースには，このほかRGBケーブルと拡張F-BASIC（テープ版）が添付される。

拡張MSX BASICによって，FM-XでFM-7のキーボード，RAM，プリンタポート，RS-232Cポートを使用することが可能になる。また，拡張F-BASICによって，FM-7でFM-XのVDP，PSG，ジョイスティックポートを使用することが可能になる。

図1　1台の家庭用TVをFM-7，Xで共用する場合（基本システム）

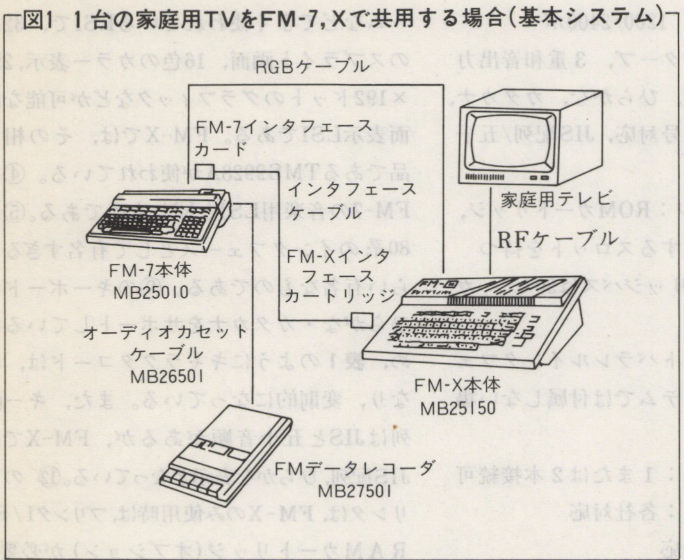
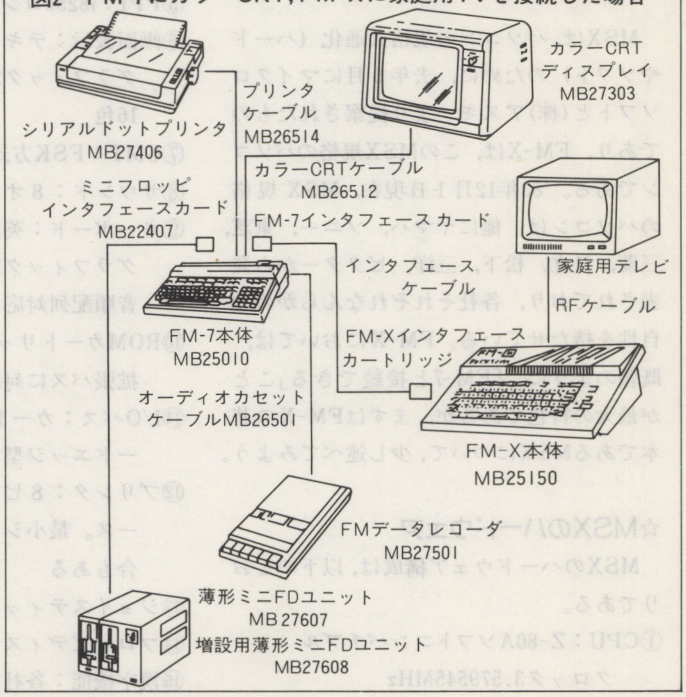


図2　FM-7にカラーCRT，FM-Xに家庭用TVを接続した場合



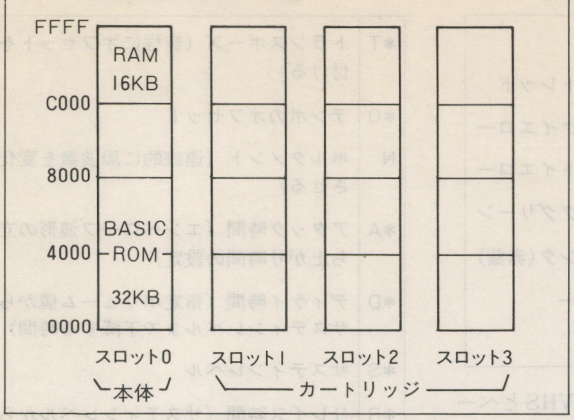


図3　FM-X標準メモリマップ

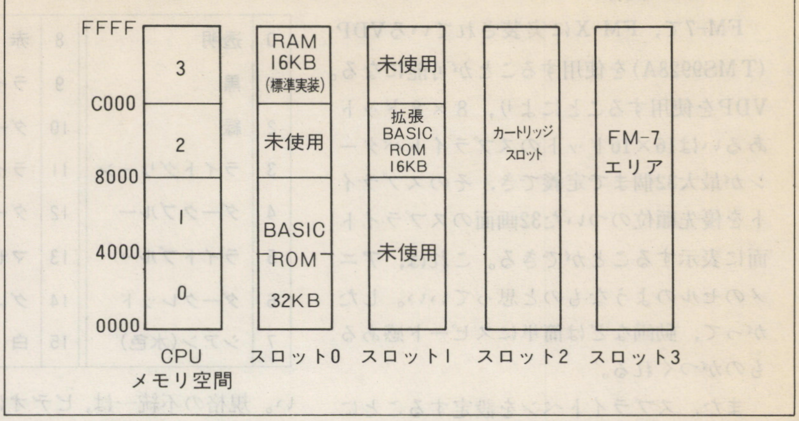


図4　拡張MSX BASIC起動時のメモリマップ

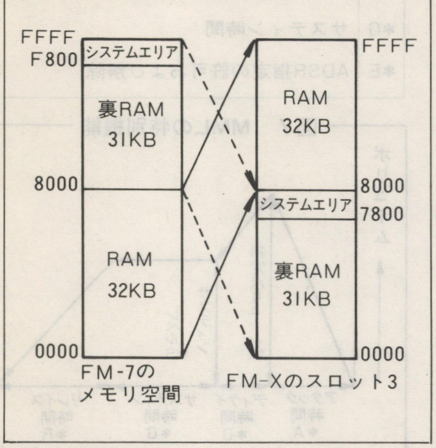


図5　FM-7エリア

☆拡張MSX　BASIC

FM-Xインタフェースカートリッジ内のROMに搭載されている拡張MSX BASICシステムは，&HFD05番地に〝1〟を書き込むことによって起動させる，という方式をとっている。

(1)　メモリの拡張

MSXでは，CPUのメモリ空間は4ページに区分され，スロットセレクトレジスタにより16Kバイトのページごとに任意のスロットから選んでメモリに配置される。図3に標準FM-Xのメモリマップを示す。FM-7を接続することにより，&H8000～&HBFFF番地がスロット3に割り当てられる(図4参照)。FM-7インタフェースでは，FM-X側のアドレスバスA15が反転してFM-7に出力されているので，メモリ空間は図5のようになる。つまり，拡張RAM16Kバイト(&H8000～&HBFFF)には，FM-7側の&H0000～&H3FFFが使用されることになる。

(2)　プリンタ関係

標準MSX BASICでは，FMシリーズのプリンタにひらがな，グラフィック記号を印字できない。拡張MSX BASICでは，PRINTERSETUPコマンドにより，印字可能となった。

(3)　RS-232C関係

FM-7用のRS-232Cカード(オプション)を用いて，RS-232C規格のインタフェースをもった周辺機器の制御や，音響カプラを用いたデータ交換などが可能になる。

回線制御関係の拡張コマンドは，以下の9つ。

_TERM, _COMINIT, ON COM GOSUB _COM (ON/OFF/STOP), OPEN, CLOSE, INPUT＃, LINE INPUT＃, PRINT＃

(4)　モニタ機能

MSX BASICには，Z-80のメモリ内容の表示などのモニタ機能がないが，拡張MSX BASICには，かなり強力なモニタ機能がある。メモリ内容の表示・変更のほか，キャラクタ表示や転送，ブレークポイントの設定などである。表2に_MONコマンドのサブコマンドを示す。

(5)　FM-7キーボードからの入力

FM-Xのキーボードには，テンキー部がない。FM-7のキーボードは，このテンキー部があるだけでなく，テンキー部に＊や／や＋，－を加えるなど，かなり使いやすくできている。そこで，FM-7のキーボードに慣れている人は，拡張MSX BASIC時にFM-7のキーボードを用いて入力することができるようになっている。もちろん，DUPキーなど一部制限はある。

☆拡張F-BASIC

F-BASIC V3.0にVDP, PSG (FM-7とFM-Xと両方の)，ジョイスティックのサポートを追加した拡張F-BASICは，FM-7の裏RAMに展開される。また，図6のように，ユーザーRAMを約500バイト追加使用することになる。拡張F-BASICはカセットテープで供給されるので，ディスクユーザーは，このテープとF-BASIC V3.0システムディスクを使って，自分で拡張F-BASICシステムディスクを作ることになる。

表2　MONコマンド

C	メモリ内容をASCIIキャラクタで表示
D	メモリ内容を16進数で表示
S	メモリ内容の変更
F	メモリブロックの内容を変更
M	メモリブロックの内容を転送
B	指定アドレスにブレークポイントを設定
G	指定アドレスから機械語プログラムの実行
X	レジスタ内容の表示，変更
I	I/O領域の指定アドレスの内容を表示
O	I/O領域の指定アドレスにデータの書き込み
Q	BASICコマンドレベルに戻す

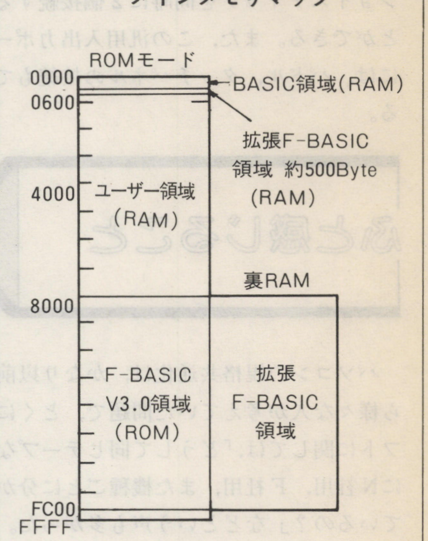


図6　拡張F-BASIC実行時のメインメモリマップ

(1) **VDP関係**

FM-7で，FM-Xに実装されているVDP(TMS9928A)を使用することが可能になる。VDPを使用することにより，8×8ドットあるいは16×16ドットのスプライトパターンが最大32個まで定義でき，そのスプライトを優先順位のついた32画面のスプライト面に表示することができる。これは，アニメのセルのようなものと思っていい。したがって，動画などは簡単にスピード感あるものがつくれる。

また，スプライトペンを設定することにより，簡単にタートルグラフィックを実現できる。

VDP関係の拡張コマンドは，CMD SCODE, CMD COLORなど38。カラーコード対応表を**表3**に示す。

(2) **PSG関係**

FM-7とFM-XのPSGを両方使用することにより，3重和音のステレオおよび6重和音のモノラル演奏が可能になる。

また，エンベロープ波形の立ち上がり時間設定や，チャンネルごとのボリューム調整などの特別機能をサポート（**表4**，**図7**参照）している。

(3) **ジョイスティック関係**

FM-X本体の汎用入出力ポートを使ってジョイスティックを同時に2個接続することができる。また，この汎用入出力ポートには，パドル，タッチパネルの接続もできる。

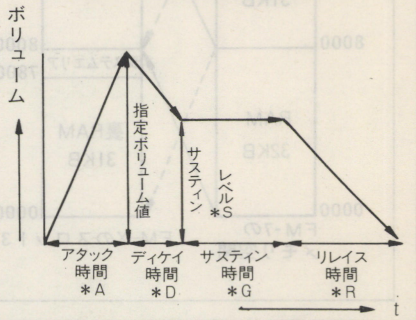
表3　カラーコード

0	透明	8	赤
1	黒	9	ライトレッド
2	緑	10	ダークイエロー
3	ライトグリーン	11	ライトイエロー
4	ダークブルー	12	ダークグリーン
5	ライトブルー	13	マゼンタ(赤紫)
6	ダークレッド	14	グレー
7	シアン(水色)	15	白

表4　MML特別機能

*T	トランスポーズ（音程にオフセットを付ける）
*Q	テンポのオフセット
N	ポルタメント（連続的に周波数を変化させる）
*A	アタック時間（エンベロープ波形の立ち上がり時間の設定）
*D	ディケイ時間（指定ボリューム値からサスティンレベルまで下降する時間）
*S	サスティンレベル
*R	リレイス時間（サスティンレベルから0まで下降する時間）
*G	サスティン時間
*E	ADSR指定の許可および解除

図7　MMLの特別機能



ふと感じること

パソコンの規格共通化は，かなり以前から様々な人が考えていた問題で，とくにソフトに関しては，「どうして同じテープなのにN社用，F社用，また機種ごとに分かれているの？」などという声も多かった。また，ファームウェア（たとえばBASICのエントリアドレス，ソースリストなど）の公開もされることが少なかった。そうした意味で，MSXという統一規格案はかなりスゴイことだと思う。

反面，いくつかの問題も出てくる。パソコンの発展は，ハードの進歩と切り離せない。規格の不統一は，ビデオにVHSとベータマックスの2方式があるのとはわけが違う。たとえばFM-7と8はほとんどコンパチではあるが，音楽やパレット命令のあるソフトはFM-8では動かない。もし，コンパチであることを絶対条件として，FM-7のPSGやパレットを削ってしまったら，7の魅力は半減する。各メーカーの価格面の競争のみならず，ハード的な仕様の競争によって，現在のパソコンがある。しかしMSXにおいては，各メーカー間の競争は本来，価格面に限られる。統一規格という前提に立てば，ハードの追究は意味がない。

現在，発表されているMSXマシンは，各社それぞれ独自の特色を出している。FM-XのFM-7との接続，ヤマハのシンセサイザとの接続，東芝の64KB RAM…などなど。しかし，これらのマシンの特色を最大限ひき出すプログラムとなると，どのマシンでもOKというわけにはいかなくなる。あまり各社のマシンの個性を強調しすぎると，MSX本来の〝共通性〟と矛盾してしまう。とはいっても，たとえばヤマハでは，MIDI規格のシンセサイザがつなげるアダプタを販売する予定だという。そうなればなにもヤマハのMSXマシンのみならず，FM-Xでもいいわけで，同じようにFM-7と接続できる他社のMSXマシンも出てくるかもしれない，という疑問も生じてくる。

また，〝どのメーカーのソフトも共通して使える〟というのも，誤解を生じやすい。これは，〝MSX規格のソフトなら〟というただし書きがつくわけで，なんだ当たり前じゃないか，という読者は多いだろうが，パソコンをまったく知らない人は，そうではない。FM-7やPC，MZなどの今までのソフトも使えると勘違いする人も多いのではないだろうか。

おわりに

ざっとMSX，そしてFM-Xについて書いてきたが，MSX自体，スプライト機能などなかなか魅力あるマシンだと思う。5万円弱から7万円台という価格を加味すれば，ゲームマシン（にしてはちょっと高いかな？）あるいはパソコンやBASICに初めてふれる人の入門マシンとしてはなかなかだと思う。それであきたらない人は，増設あるいは上位のマシンへと進めばいいだろう。

MSXの統一規格が提唱されて半年。これからの動向など，かなり興味のあるところだ。また，基本コマンドは同じなので，どのメーカーのマニュアルが一番親切でわかりやすいか，などやじうま的興味も生じてくる。

*　　　*　　　*

参考文献

- ●ASCII Vol.7　#8
- ●MSXテクニカルデータブック[1]　アスキー
- ●富士通FM-Xユーザーズマニュアル

新連載

マシン語ゲームの作り方

〔1〕

笠作貴弥　小牧徳夫

最近の世の中には，(原田知世とか小泉今日子などのアイドルスターや青木とか中畑などのスタープレイヤーなんていう)スターと名のつく人がたくさんいて，あぁ！自分もスターになれたらな〜と思う日も時々あります。そんな中，パソコン界にもスタープログラマなんていう人達が登場して，いまやソフトウェア業界を10本の指で担っています。

“この道なら，なんとか自分でもやれるのでは”と単純に，いやマジにそう思う人もいるでしょう。けれど，さあプログラムを作ろうかと思い立ってはみるものの，今やゲームのほとんどがマシン語で走っている。マシン語と聞いたとたんに，その決意が様々に変わってしまう。よし，マシン語でゲームを作ったろかという人も，マシン語なんて難しいのよね〜という人も，マシン語ぐらい使えるわいという人も，まあお付き合いしてみてください。

キャラクタ移動や画面構成などの基本テクニックから，キャラクタ移動中に効果音を発生させるとか，個々のキャラクタに速度差をつけて個性をもたせるなんていう味のあるテクニックまで今回から数回にわたって話してみようと思います。

何はともあれ，パソコンの楽しみは，自分で作ること，自分なりに味付けのできることです。スターになるには地道な努力も必要なんです。　(N. K)

×　◆　×

I　ゲーム作りは，アイデアから……

何をするにもきっかけというものが必要ですが，ゲーム作りの場合は，アイデア，そう１つのアイデアから始まるのです。

アイデアといっても，例えば，高速スクロールルーチンを思いついたとか，マシン語で迷路を作る方法を思いついたとかいうプログラムテクニック上のものや，好きな映画，詩や歌の内容をゲーム化したいというような，ストーリ上のもの，またかわいいキャラクタを思いついたというようなデザイン上のものなど，いろいろです。

要するに，何かひらめいて，それをゲームに使いたい，応用したいと思えばよいのです。

どうですか？　何かひらめきましたか？そうしたら次は，シナリオ作りです。

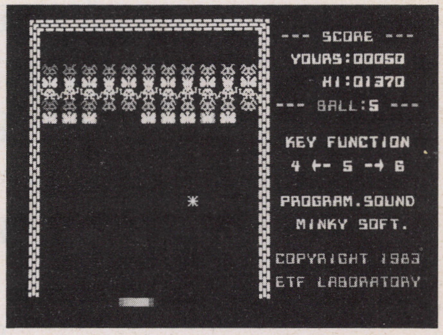
II　アイデアを生かすも殺すもシナリオ次第

アイデアから即プログラム作りに入ってもいいのですが，そこは，はやる気持ちを抑えて，じっくりとアイデアをふくらませて，シナリオ作りをしましょう。

余談になりますが，あの大ヒット中のゲーム「ゼビウス」は，画面がスクロールするようなゲームという１つのアイデアから，3部にわたる長編のシナリオを書き，制作されたそうです。

では，具体的にシナリオを書いてみましょう。

ここでは，「ブロックくずし」のゲームを



１つのアイデアとして，逆にそこからシナリオを書いて，またゲームとして再構成するようにしてみます。

◇

ある日のこと，レプリカ星のクリン君の家の一室に，インベ虫の大群が発生しました。この虫，なんの害もありませんが気味が悪い。隣のスクエア星では気味の悪さにショック死した人までいるそうです。

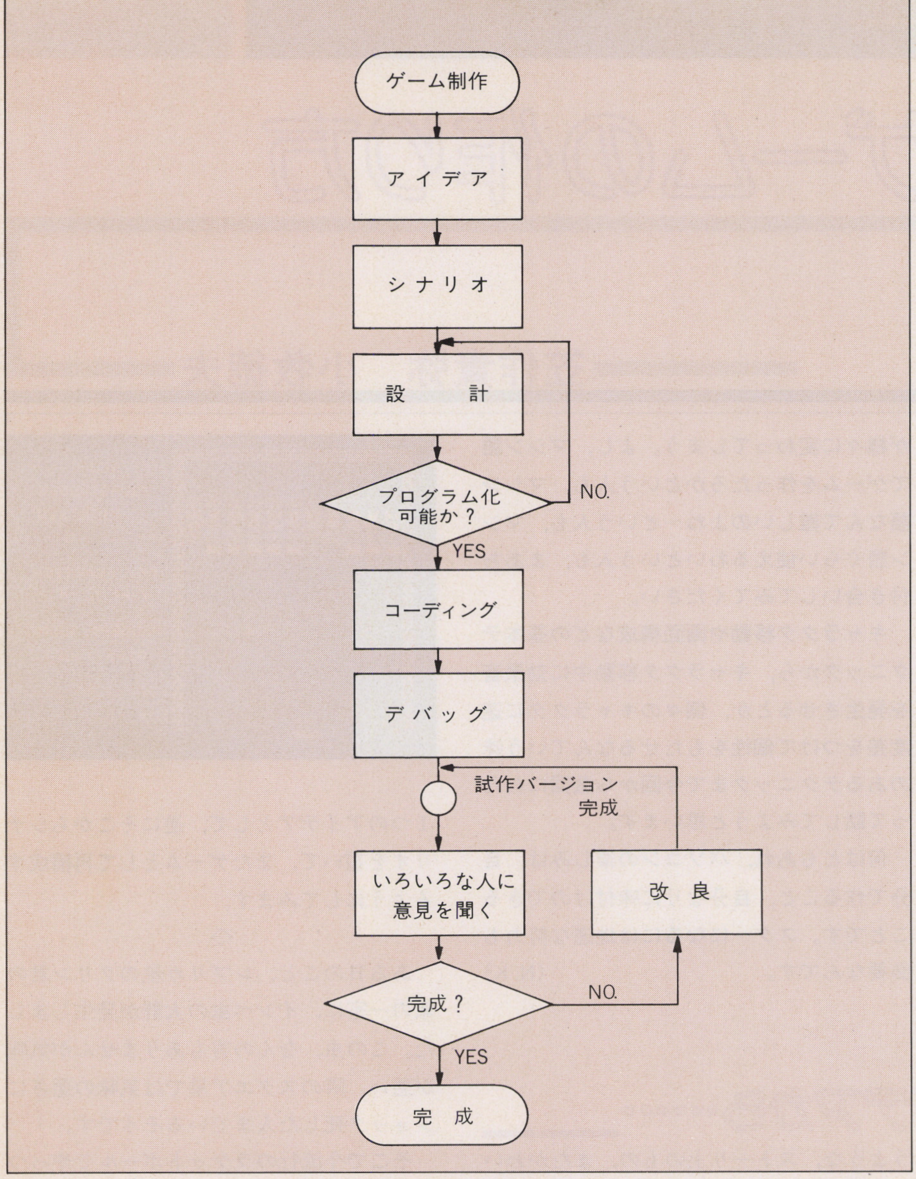
そこで今流行のラケットボールでやっつけることにしました。あいにくこの部屋，四方全部が壁でなく，一方が大きな窓になっています。

インベ虫をやっつけたら，おとうさんにほめられることは言うまでもありませんが，しかしせっかくの手柄も窓を壊してしまったらなんにもなりません。ボールが窓に当たらないように気をつけて，さあラケットボールというよりインベ崩しの始まりです。今日のクリン君のおやつが2倍になりますように。

◇

どうですか？　かなり翔んでいるような気がしますが，これぐらい派手にしておくと，あとのプログラム作りにも意欲がわいてくるし，最終的な作品が，一味も二味も違ってくるものなのデス。

図1　ゲーム作りの手順



III 設　計

シナリオが完成したら，いよいよそれをプログラム化するために細かな設計に入ります。

ここでいう設計とは，画面構成や，キャラクタの設定，各キャラクタの移動のアルゴリズム，得点の計算方法というようなゲーム全体にわたる約束事を，シナリオに従って決めることです。

では，前項で示したブロックくずしを例にとって，具体的な話を始めたいと思いますが，その前に画面構成などを考える場合，ターゲットになるマシンのハード上の特性を充分に知っていないと設計ができません。

そこで，本題からは，少しはずれて，先にFM-7のハードウェア上の特性について説明します。本稿は，一応FM-7を中心に，FM-8，11は変更点を次号以降に示すという形で進めていきます。

○キー入力

FM-7では，メインシステムの$FD01あるいは，サブシステムの$D401を参照することで，現在押されているキーまたは，以前押されていたキーのキャラクタコードが読み出せます。

この場合，以前押されていたというのが曲者で，FM-7の場合，新しいキーを押さない限り，キーを離しても前に押したキーのデータが残ったままになってしまうのです。これは，設計上，非常に重要なことで，このことをよく頭に置いておかないとあとでひどい目にあいます。

それから，これもハードの特性上のことですが，同時に2つ以上のキーをセンスできません。ですから，キーを同時に2つ以上押すゲームは，FM-7では，不可能です。ただしFIRQを禁止して$FD04を参照すればBREAKキーの状態を知ることができるので，これもキーの1つとして使えます。

○サブシステム

FM-7では，画面表示のほとんど全て(パレット機能などはメイン側）をサブCPUが管理しています。ですから，高速のリアルタイムゲームなどを制作するためには，完全にサブシステムを理解していなければなりません。

サブシステムへプログラムを転送，実行する方法は，次回詳しく説明することにして，今回は，VRAMと各種I/O，共有RAMについて取り上げます。

○VRAM

VRAMと実際の画面の対応は，図3のようになっています。つまり，0番地に$80を書き込めば，（0，0）の座標に青い点が打たれるわけです。同様に$4000，$8000番地に$80を書くと（0，0）に黄色の点が打たれます。

また，x方向に関しては，ビット単位の対応となっているので，キャラクタなどを移動する場合は，x方向には，8ドット（バイト）単位で動かすのが簡単です。

○I/O

1）$D400，01　キー入力

図2　サブシステムのメモリマップ（FM-7，8）

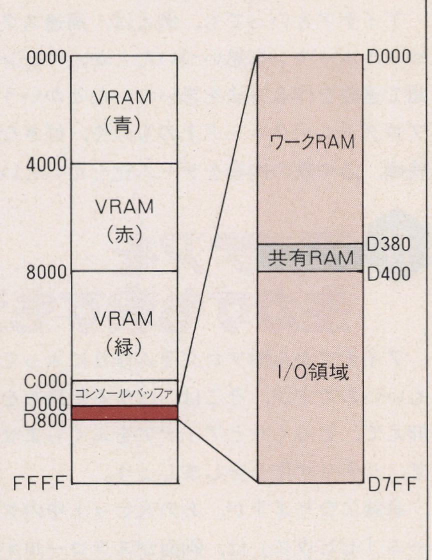


図3

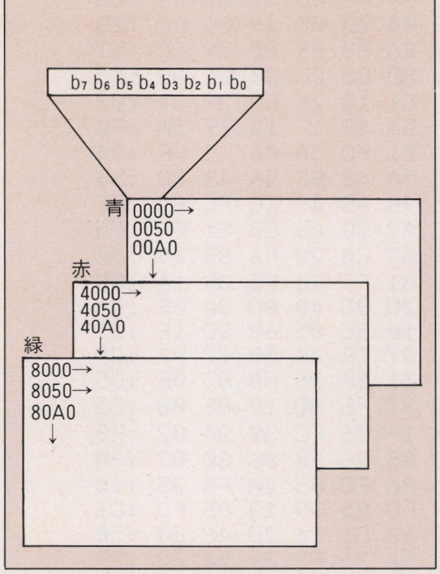


図4　I/O領域（FM-7,8）

アドレス	意　　味	リード	ライト
D400,01	KEY DATA		×
D402	CANCEL IRQのアクノリッジ		×
D403	BUZZER		×
D404	MAIN CPUへ割り込み		×
D408	CRTのON/OFF	ON	OFF
D409	VRAMアクセスフラグ	SET	RESET
D40A	BUSYフラグ	READY	BUSY
D40D	INS LEDのON/OFF	ON	OFF
D40E,0F	VRAMのオフセット	禁止	

キーデータを参照できます。なお，FM-7ではメイン側からも読みだせます。

2）$D402

メイン側からの割り込み要求を受け付けたことをメイン側に知らせるのに用います。

3）$D403　ベル

このアドレスをリードするとベルが1回鳴ります。

4）$D404

メインCPUへ割り込み要求をする場合に用います。

5）$D408

CRT画面の表示をコントロールします。

6）$D409　VRAMアクセスフラグ

CRTコントローラとCPUのVRAMへのアクセスが競合しないように設定するものです。このアドレスをセットすると，CRTCがVRAMをアクセスするたびにサブCPUをHALTします。ですから，あまり速度を要求されない場合は，セットしたままでもよいのですが，VRAMをCPUがアクセスするたびにセット，リセットを繰り返したほうが数倍速くなります。

この際，注意してほしいのは，CPUとCRTCのアクセスは，非同期で動作しているので，確実に競合を防ぐには，アクセスフラグをセットしたあと，VRAMをアクセスしない命令を1つ入れる必要があります。

7）$D40D

INS LEDを点滅させます。

8）$D40E,0F

VRAMにオフセットを与えるためのエリアです。このポートをうまく利用するとハード的に2ドット単位で上下スクロールが，できます。

○共有RAM

メインCPUとサブCPU双方からアクセスすることができるRAM領域のことです。

メインCPUとサブCPUのデータ通信は全てこの領域を通して行います。

マッピングは，サブCPU側で$D380～$D3FF，メイン側では$FC80～$FCFFで，それぞれ$D380と$FC80，$D381と$FC81というように対応しています。

共有RAMをアクセスする場合は，メインCPUとサブCPUのアクセスが競合しないように次の手順で行います。

a）メインCPUからアクセスする場合

○サブシステムがREADY状態であるか調べる

○サブCPUをHALTする

○サブCPUがHALTしたことを確認する

○アクセス可能

b）サブCPUからアクセスする場合

メイン側で，競合を防ぐためにサブCPUをコントロールしているので，いつでもアクセス可能です。

今回は，ゲーム作りの手順とサブシステムの一部について書きましたが，サブシステムの部分は，初心者の方には少しわかりづらかったと思いますが ゲーム作りを究極的にするためには，ハードの知識は，必要不可欠です。ですから，ガンバッて理解してください。

次回は，サブシステムの残りの部分と本格的な設計に入っていきます。

なお，今回のシナリオに従って作ったプログラムリストを**リスト1**に示します。FM-8,11用およびソースリストは，次回以降，解説の進行に従って掲載していきます。

リスト1　（FM-7用，入口番地$5000）

©1983　ETF Laboratory

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
5000  10 CE FC 7F 86 FC 1F 8B :85
5008  17 03 74 8E 59 82 17 03 :11
5010  52 17 00 90 10 CE FC 7F :52
5018  17 00 87 8E 5A 7D 17 03 :1D
5020  42 86 04 B7 59 71 8E 0B :E6
5028  B8 34 10 17 02 BF 8D 28 :89
5030  35 10 30 1F 26 F3 8D 6A :A4
5038  17 00 99 8E 5A DB 17 03 :8D
5040  22 17 02 A9 17 00 FA 8D :82
5048  0F 17 01 48 17 02 02 B6 :40
5050  59 7B 81 23 27 BE 20 E9 :66
5058  1A 40 B6 FD 04 85 02 27 :BF
5060  01 39 B6 FD 01 10 CE FC :C8
5068  7F 86 04 B7 FD 02 86 0C :51
5070  B7 59 71 86 03 B7 59 73 :8D
5078  1C EF 8D 26 8E 5A EE 17 :AB
5080  02 E1 8D 50 8E 01 90 8D :6C
5088  6F 8E 5A FA 17 02 D4 8D :CB
5090  43 17 02 59 17 02 2A 17 :0F
5098  00 A7 17 01 C6 17 00 F4 :90
50A0  20 EF 8D 20 8E 18 00 CE :30
50A8  5B 7E C6 37 F7 59 7B A6 :47
50B0  C0 A7 80 5A 26 F9 17 02 :79
50B8  34 1C AF 8E 5B 15 17 01 :15
50C0  EE 16 02 62 17 02 E7 86 :EE
50C8  E0 97 82 7F FD 05 86 0A :0A
50D0  B7 59 7D 39 8E 00 00 BF :13
50D8  59 76 8D 28 8D 2E C6 06 :0B
50E0  F7 59 7A F6 59 7A 86 07 :20
```

```
50E8  1F 01 CC 45 07 17 02 57 :A8
50F0  86 09 B7 59 7D 8E 01 00 :AB
50F8  34 10 17 01 F0 35 10 30 :C1
5100  1F 26 F5 39 BE 59 76 CC :CC
5108  42 03 20 06 BE 59 78 CC :C6
5110  42 05 FD 5A 4A 1F 10 10 :27
5118  8E 5A 4E 8E 59 63 EE 81 :EF
5120  27 19 34 40 CE 00 00 A3 :25
5128  E4 25 04 33 41 20 F8 E3 :7C
5130  E1 34 06 1F 30 E7 A0 35 :26
5138  06 20 E3 8E 5A 4A 16 02 :53
5140  22 FC 59 7C BB 59 7E 27 :AC
5148  33 81 16 22 2F FB 59 7F :EE
5150  C1 04 25 28 C1 08 22 24 :21
5158  8D 23 A6 C4 27 1E 6F C4 :92
5160  C6 0A 3D F3 59 76 FD 59 :25
5168  76 8D 99 70 59 7F 8E 5B :CD
5170  06 17 01 3B 7A 59 7B 26 :CD
5178  03 17 FF 26 39 34 06 86 :38
5180  0B C0 04 3D 34 06 E6 62 :8E
5188  4F 5A 54 E3 E1 CE 18 00 :A7
5190  33 CB 35 86 7A 59 72 10 :0E
5198  26 00 A9 B6 59 71 B7 59 :5F
51A0  72 FC 59 7C 17 00 A6 FD :FD
51A8  5A 55 8E 5A 55 17 01 B3 :B7
51B0  FC 59 7C 81 01 22 05 86 :00
51B8  01 B7 59 7E 81 16 25 05 :50
51C0  86 FF B7 59 7E C1 01 26 :FB
51C8  0D 86 06 B7 59 71 86 02 :A2
51D0  B7 59 73 F7 59 7F C1 17 :2A
51D8  26 2B B6 59 80 B1 59 7C :66
51E0  27 10 4C B1 59 7C 27 0F :3F
51E8  4C B1 59 7C 26 17 86 01 :96
51F0  20 02 86 FF B7 59 7E 70 :A5
51F8  59 7E 86 FF B7 59 7F 8E :79
5200  5B 66 17 00 AA C1 18 26 :81
5208  1C 5A B6 59 80 B1 59 7C :8B
5210  27 E0 8B 02 B1 59 7C 27 :41
5218  D5 7A 59 7A 8E 5B 72 17 :94
5220  00 8D 17 FE BE FC 59 7C :31
5228  BB 59 7E FB 59 7F 81 01 :E7
5230  25 13 81 16 22 13 FD 59 :5A
5238  7C 8D 12 FD 5A 5C 8E 5A :B6
5240  5C 17 01 1F 39 86 01 20 :73
5248  ED 86 16 20 E9 48 8B 02 :67
5250  39 B6 59 7C B7 59 80 48 :9C
5258  80 02 B7 5A 63 8E 5A 63 :41
5260  16 01 00 7A 59 74 26 40 :C4
5268  B6 59 73 B7 59 74 17 01 :1E
5270  3D 86 E1 97 82 7F FD 05 :3E
5278  12 12 12 17 01 30 86 E4 :E8
5280  97 82 96 83 7F FD 05 81 :34
5288  36 27 08 81 34 26 19 86 :DF
5290  FF 20 02 86 01 BB 59 81 :3D
5298  81 02 25 0C 81 2C 22 08 :8B
52A0  B7 59 81 44 4A B7 59 80 :AF
52A8  B6 59 81 80 02 20 AB 34 :11
52B0  08 7F 03 13 4F 1F 8B BF :55
52B8  00 D9 BD EC 72 1A 40 35 :83
52C0  88 7D 59 7A 26 26 BE 59 :3B
52C8  76 BC 59 78 25 03 BF 59 :43
52D0  78 8E 5A 6E 17 00 8C 8E :FF
52D8  5B 50 8D D3 4F 34 02 8D :1D
52E0  0C 35 02 4A 26 F7 17 FD :BE
52E8  DB 16 FD 28 39 7A 59 70 :92
52F0  26 33 B6 59 6F B7 59 70 :57
52F8  17 00 B3 CE 18 00 8E FC :3A
5300  82 86 E3 A7 80 C6 37 73 :82
5308  59 75 A6 C4 27 0D 7D 59 :42
5310  75 4A 47 24 03 48 20 02 :97
5318  48 4C 4C A7 C0 A7 80 5A :C8
```

```
5320  26 E8 7F FD 05 39 8E 05 :5B
5328  28 CC 04 00 8D 19 8B 02 :2B
5330  81 30 26 F8 8E 05 29 CC :57
5338  02 01 8D 03 CC 30 01 8D :1D
5340  06 5C C1 18 26 F9 39 34 :C7
5348  16 FD 5A 43 1F 10 B7 5A :F0
5350  45 86 01 FD 5A 46 CC FF :34
5358  FF FD 5A 48 8E 5A 43 8D :56
5360  02 35 96 8D 49 CE FC 82 :EF
5368  86 E2 A7 C0 A6 80 81 FF :75
5370  27 04 A7 C0 20 F6 8D 49 :7E
5378  A6 84 81 FF 26 E5 39 1A :08
5380  50 8D 2B 8D 40 8D 3A 8E :2A
5388  54 03 10 8E 05 60 8D 1E :05
5390  86 02 97 FE A6 80 97 FF :D9
5398  8D 27 31 3F 26 F0 8D 0E :D5
53A0  86 01 97 FE 8D 1B 8E 00 :52
53A8  00 30 1F 26 FC 39 34 02 :E0
53B0  B6 FD 05 2B FB 86 80 B7 :9B
53B8  FD 05 B6 FD 05 2A FB 35 :14
53C0  82 7F FD 05 39 10 8E FC :D6
53C8  80 8E 53 D6 C6 2D A6 80 :50
53D0  A7 A0 5A 26 F9 39 00 00 :F9
53D8  3F 59 41 4D 41 55 43 48 :47
53E0  49 93 D3 8F 90 8E C0 00 :1C
53E8  7F D3 FE B6 D4 0A B6 D3 :6D
53F0  FE 27 FB B7 D4 0A 4A 27 :26
53F8  07 B6 D3 FF A7 80 20 E8 :BE
5400  7E C0 00 10 CE CF FF 17 :01
5408  00 97 7F D3 82 B5 D4 0D :01
5410  B6 D4 0A B6 D3 82 27 FB :C1
5418  B7 D4 0A B7 D4 0D 80 E0 :8D
5420  48 30 8C 06 EC 86 AD 8B :B4
5428  20 E0 00 8F 01 4B 00 29 :04
5430  00 D0 00 28 86 A0 F6 D3 :E7
5438  84 3D 58 49 58 49 34 06 :3D
5440  F6 D3 83 1D E3 E1 1F 01 :4D
5448  39 C6 10 3D 33 8D 02 13 :21
5450  33 CB 39 10 8E D3 87 7D :AC
5458  D3 86 27 F6 7A D3 86 8D :D6
5460  D3 7C D3 83 7C D3 83 A6 :1D
5468  A0 8D DE 8D 02 20 E8 34 :D6
5470  20 10 8E 00 08 B6 D3 85 :D4
5478  34 02 EC C1 B5 D4 09 64 :D9
5480  E4 24 02 ED 84 64 E4 24 :E7
5488  04 ED 89 40 00 64 E0 24 :22
5490  04 ED 89 80 00 B7 D4 09 :8E
5498  30 88 50 31 3F 26 D6 35 :A9
54A0  A0 B7 D4 08 B5 D4 09 4F :14
54A8  5F 1F 01 ED 81 8C C0 00 :39
54B0  26 F9 B7 D4 09 B6 D4 08 :45
54B8  39 8E 02 9A CE 02 99 34 :00
54C0  50 C6 C0 B5 D4 09 6F 84 :5B
54C8  6F 89 40 00 6F 89 80 00 :B0
54D0  6F C4 6F C9 40 00 6F C9 :E3
54D8  80 00 30 88 50 33 C8 50 :D3
54E0  86 32 4A 26 FD 5A 26 DE :83
54E8  B7 D4 09 35 50 30 01 33 :7D
54F0  5F 8C 02 B0 26 C9 39 D3 :98
54F8  83 00 10 AE 8C F9 E6 8C :38
5500  F8 4F 8D 19 4C 81 0B 26 :EB
5508  F9 5C E7 8C EC C1 05 26 :A0
5510  07 6F 8C E5 10 8E D3 83 :DB
5518  10 AF 8C DB 39 34 06 86 :1F
5520  A0 3D 58 49 58 49 34 06 :59
5528  E6 62 58 58 4F E3 E1 8E :99
5530  0A 04 30 8B A6 A0 C6 21 :F6
5538  3D 33 8D 00 3F 33 CB A6 :E0
5540  C0 C6 08 8D 0A 8D 08 30 :EA
5548  88 4C 5A 26 F6 35 86 34 :39
5550  06 34 02 EC C1 B5 D4 09 :7B
```

```
5558  64 E4 24 02 ED 84 64 E4 :27
5560  24 04 ED 89 40 00 64 E0 :22
5568  24 04 ED 89 80 00 30 02 :50
5570  B7 D4 09 35 86 B6 D4 01 :DA
5578  B7 D3 83 39 07 00 00 00 :4D
5580  00 00 00 00 00 00 00 00 :00
5588  00 00 00 00 00 00 00 00 :00
5590  00 00 00 00 00 00 00 00 :00
5598  00 00 00 00 00 05 00 01 :06
55A0  80 00 07 07 E0 E0 0F C1 :1E
55A8  83 F0 0F F1 8F F0 03 F9 :EE
55B0  9F C0 0F FF FF F0 0F F9 :64
55B8  9F F0 07 E1 87 E0 05 00 :E3
55C0  04 20 00 00 C2 43 00 01 :2A
55C8  F1 8F 80 00 FD BF 00 00 :BC
55D0  3F FC 00 00 FF FF 00 00 :39
55D8  7D BE 00 00 39 9C 00 02 :12
55E0  03 E0 07 C0 00 70 0E 00 :28
55E8  00 3B DC 00 00 7F FE 00 :94
55F0  01 E3 C7 80 07 83 C1 E0 :56
55F8  07 FF FF E0 00 F0 0F 00 :E4
5600  02 01 C0 03 80 07 F0 0F :4C
5608  E0 0E 3B DC 70 18 7F FE :0A
5610  18 01 E3 C7 80 07 83 C1 :8E
5618  E0 07 FF FF E0 00 3C 3C :3D
5620  00 06 00 1F F8 1E 00 7F :BA
5628  FE 0C 00 E3 C7 0C 1F FC :DB
5630  3F FC 31 FF FF 80 60 0C :56
5638  30 00 F0 0C 30 00 00 FC :58
5640  3F 00 06 00 00 00 00 78 :BD
5648  1F F8 00 30 7F FE 00 30 :F4
5650  E3 C7 00 3F FC 3F F8 01 :1D
5658  FF FF 8C 00 0C 30 06 00 :CC
5660  FC 3F 0F 00 00 3F FC 30 :B5
5668  0C 30 0C 30 3C 30 3C 30 :50
5670  3C 3F FC 00 00 07 80 01 :FF
5678  80 01 80 07 80 07 80 07 :16
5680  80 07 80 00 00 3F FC 3C :7E
5688  0C 00 0C 3F FC 3C 00 3C :CB
5690  3C 3F FC 00 00 3F FC 00 :B2
5698  0C 00 0C 0F FC 00 3C 3C :9B
56A0  3C 3F FC 00 00 06 30 0C :B9
56A8  30 18 30 3F FC 00 F0 00 :A3
56B0  F0 00 F0 00 00 3F FC 30 :4B
56B8  00 30 00 3F FC 00 3C 3C :E3
56C0  3C 3F FC 00 00 3F F8 30 :DE
56C8  78 30 00 3F FC 3C 0C 3C :67
56D0  0C 3F FC 00 00 3F FC 3C :BE
56D8  1C 00 18 00 F0 03 C0 03 :EA
56E0  C0 03 C0 00 00 0F F0 0C :8E
56E8  30 0C 30 3F FC 3C 0C 3C :2B
56F0  0C 3F FC 00 00 3F FC 30 :B2
56F8  0C 30 0C 3F FC 00 3C 3C :FB
5700  3C 3F FC 00 00 0F FC 0C :8E
5708  0C 0C 0C 3F FC 3C 0C 3C :E3
5710  0C 3C 0C 00 00 3F F0 30 :B3
5718  30 30 30 3F FC 30 3C 30 :67
5720  3C 3F FC 00 00 3F FC 30 :E2
5728  3C 30 00 3C 00 3C 00 3C :20
5730  0C 3F FC 00 00 3F F0 30 :A6
5738  18 30 0C 3C 0C 3C 0C 3C :20
5740  18 3F F0 00 00 3F F0 30 :A6
5748  00 30 00 3F F0 3C 00 3C :D7
5750  0C 3F FC 00 00 3F F8 30 :AE
5758  00 30 00 3F E0 3C 00 3C :C7
5760  00 3C 00 00 00 3F FC 30 :A7
5768  3C 30 00 3C 3C 3C 0C 3C :68
5770  0C 3F FC 00 00 0C 0C 0C :6B
5778  0C 0C 0C 3F FC 3C 0C 3C :E3
5780  0C 3C 0C 00 00 00 C0 00 :14
5788  C0 00 C0 03 C0 03 C0 03 :09
5790  C0 03 C0 00 00 00 30 00 :B3
5798  30 00 30 00 3C 00 3C 30 :08
57A0  3C 3F FC 00 00 30 78 31 :50
57A8  C0 37 00 3F FC 3C 0C 3C :B6
57B0  0C 3C 0C 00 00 0C 00 0C :6C
57B8  00 0C 00 3C 00 3C 00 3C :C0
57C0  00 3F FC 00 00 3C 3C 36 :E9
57C8  6C 33 CC 31 8C 30 0C 3C :A0
57D0  0C 3C 0C 00 00 0C 0C 0F :7B
57D8  0C 0D 8C 3C CC 3C 6C 3C :91
57E0  3C 3C 0C 00 00 3F FC 30 :EF
57E8  0C 30 0C 30 3C 30 3C 30 :50
57F0  3C 3F FC 00 00 3F FC 30 :E2
57F8  0C 30 0C 3F FC 3C 00 3C :FB
5800  00 3C 00 00 00 3F FC 30 :A7
5808  0C 30 0C 3C 0C 3C 6C 3C :74
5810  30 3F CE 00 00 3F F0 30 :9C
5818  30 30 30 3F FC 3C 0C 3C :4F
5820  0C 3C 0C 00 00 3F F0 30 :B3
5828  F0 30 00 3F FC 00 3C 00 :97
5830  3C 3F FC 00 00 3F FC 01 :B3
5838  80 01 80 01 E0 01 E0 01 :C4
5840  E0 01 E0 00 00 30 0C 30 :2D
5848  0C 30 0C 3C 0C 3C 0C 3C :14
5850  0C 3F FC 00 00 30 3C 18 :CB
5858  3C 0C 0C 06 0C 03 0C 01 :76
5860  98 00 F0 00 00 30 3C 30 :24
5868  3C 30 0C 31 8C 33 CC 36 :6A
5870  6C 3C 1C 00 00 3C 0C 3C :48
5878  0C 0C 30 03 C0 0C 30 30 :77
5880  3C 30 3C 00 00 30 1E 30 :26
5888  1E 18 18 07 E0 03 C0 03 :FB
5890  C0 03 C0 00 00 3F FC 3C :FA
5898  30 00 C0 3F F0 06 00 0C :31
58A0  3C 3F FC 00 00 00 00 00 :77
58A8  00 00 00 00 00 00 00 00 :00
58B0  00 00 00 00 00 03 C0 03 :C6
58B8  C0 00 00 00 00 03 C0 03 :86
58C0  C0 00 00 00 00 00 00 00 :C0
58C8  00 00 00 00 00 00 00 03 :03
58D0  C0 03 C0 00 00 00 00 00 :83
58D8  00 1F F8 1F F8 00 00 00 :2E
58E0  00 00 00 FC 3F FC 3F 00 :76
58E8  00 3F FC 3F FC 00 00 FC :72
58F0  3F FC 3F F0 0F F3 CF F3 :2E
58F8  CF 03 C0 03 C0 F3 CF F3 :0A
5900  CF F0 0F FC 3F FC 3F F0 :34
5908  00 F3 FC 03 FC F3 C0 F3 :94
5910  CF F0 0F FC 3F FC 3F 00 :44
5918  0F 3F C0 3F C0 03 CF F3 :D2
5920  CF F0 0F 04 00 0C 00 1C :FA
5928  00 3F FC 3F FC 1C 00 0C :9E
5930  00 04 00 00 20 00 30 00 :54
5938  38 3F FC 3F FC 00 38 00 :E6
5940  30 00 20 00 00 E1 87 19 :D1
5948  98 07 E0 FF FF 07 E0 19 :7D
5950  98 E1 87 FF FF FF FF FF :FB
5958  FF FF FF FF FF FF FF 00 :F9
5960  00 00 00 27 10 03 E8 00 :22
5968  64 00 0A 00 01 00 00 08 :77
5970  08 0C 0C 03 03 00 00 00 :26
5978  00 00 00 00 01 0A 01 01 :0D
5980  0B 16 36 03 07 06 22 18 :A1
5988  1E 1B 1C 25 FF 3C 05 07 :C1
5990  03 11 12 25 FF 36 0C 07 :93
5998  0B 04 24 2C 27 24 05 24 :D3
59A0  27 2D 24 06 FF 34 0F 05 :C5
59A8  0D 19 1B 18 10 1B 0A 16 :A4
59B0  26 1C 18 1E 17 0D FF 37 :D2
59B8  11 06 0B 16 12 17 14 22 :97
59C0  24 1C 18 0F 1D 26 FF 34 :DD
```

```
59C8  01 02 0D 27 27 27 24 24 :CD
59D0  24 24 24 24 24 27 27 27 :29
59D8  FF 33 07 02 0E 27 27 27 :BE
59E0  24 24 24 24 24 25 24 24 :21
59E8  27 27 27 FF 35 0A 03 0C :C2
59F0  14 0E 22 24 0F 1E 17 0C :B8
59F8  1D 12 18 17 FF 3C 01 06 :A0
5A00  05 1C 0C 18 1B 0E FF 3B :A8
5A08  07 04 04 0B 0A 15 15 FF :4D
5A10  33 14 04 0E 0C 18 19 22 :B8
5A18  1B 12 10 11 1D 24 01 09 :99
5A20  08 03 FF 33 16 04 0E 0E :73
5A28  1D 0F 24 15 0A 0B 18 1B :AD
5A30  0A 1D 18 1B 22 FF 02 00 :7D
5A38  05 01 2A FF 30 00 05 01 :65
5A40  2B FF FF 00 00 00 00 00 :29
5A48  00 00 00 00 07 05 00 00 :0C
5A50  00 00 00 FF FF 00 00 06 :04
5A58  01 24 FF FF 00 00 06 01 :2A
5A60  2E FF FF 00 18 04 05 24 :71
5A68  2F 2F 2F 24 FF FF 11 0B :CB
5A70  06 09 10 0A 16 0E 24 18 :89
5A78  1F 0E 1B FF FF 0D 02 04 :59
5A80  0D 20 0E 24 0A 1B 0E 24 :B6
5A88  0A 15 12 0E 17 1C FF 0D :7E
5A90  0B 06 0D 11 12 1D 24 0B :8D
5A98  1B 0E 0A 14 24 14 0E 22 :AF
5AA0  FF 12 0D 06 08 1D 18 24 :85
5AA8  1C 1D 0A 1B 1D FF 0A 11 :95
5AB0  07 10 19 1B 18 10 1B 0A :98
5AB8  16 24 0D 0E 1C 12 10 17 :AA
5AC0  0E 0D FF 08 13 07 12 0B :59
5AC8  22 24 1D 0A 14 0A 16 12 :B3
5AD0  24 14 0A 1C 0A 1C 0A 14 :A2
5AD8  1E FF FF 0C 02 01 0D 0D :45
5AE0  0E 16 18 17 1C 1D 1B 0A :B1
5AE8  1D 12 18 17 FF FF 16 0C :7E
5AF0  07 06 1B 0E 0A 0D 22 26 :95
5AF8  FF FF 16 0C 07 06 24 24 :75
5B00  24 24 24 24 FF FF 22 54 :04
5B08  31 30 30 4F 35 4C 33 32 :C6
5B10  43 45 47 22 00 22 54 31 :98
5B18  35 30 4F 35 43 32 45 38 :DB
5B20  52 38 47 38 52 38 4F 34 :16
5B28  42 34 2E 4F 35 43 31 36 :D2
5B30  44 31 36 43 32 22 2C 22 :90
5B38  54 31 35 30 4C 38 43 47 :F8
5B40  45 47 43 47 45 47 44 47 :2D
5B48  46 47 43 47 45 47 22 00 :C5
5B50  22 54 31 30 30 4C 31 36 :BA
5B58  4F 33 43 4F 32 42 41 47 :10
5B60  46 45 44 43 22 00 22 54 :AA
5B68  31 30 30 4F 36 43 33 32 :BE
5B70  22 00 22 54 31 30 30 4F :78
5B78  32 43 38 2E 22 00 03 04 :04
5B80  03 04 03 04 03 04 03 04 :1C
5B88  03 01 02 01 02 01 02 01 :0D
5B90  02 01 02 01 05 06 05 06 :1C
5B98  05 06 05 06 05 06 05 04 :2A
5BA0  03 04 03 04 03 04 03 04 :1C
5BA8  03 04 02 01 02 01 02 01 :10
5BB0  02 01 02 01 02 00 00 00 :08
```


UE URBAN

アーバン オリジナルソフト

FBCP/MVI.O

FM-7・8・11用DISK版 ￥19,800（※2ドライブ必要）

F-BASIC↔CP/M ファイル転送プログラム

- F-BASIC上で使用
- 150クラスタ300KまでBASICTEXT及びシーケンシャル・ランダムファイル（データ） 全ファイルをF-BASICからCP/MディスクへCP MからF-BASICディスクへ1クラスタ（2K）単位の高速転送。
- CP/MのファイルNAMEのチェック、DISKのSETmissのチェック等ミスオペレーションによる 事故のない様に万全のチェックが施してあります。
- CP/MディスクからF-BASICディスクへの転送の際、その都度F-BASICディスクのFATを退避させていますので停電等万が一の事故にも対処できます。

クロス リファレンス リスター

FLEX CP/M-80版，5インチDISK版……￥7,800—

基本仕様

指定されたテキストファイルから先頭が英字で、次に続く文字が英数字又は'_'である単語を抜き出し、その出現回数と出現行を出力致します。

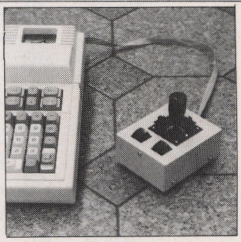
■ オプション機能

- ○英大文字・小文字の変換ができる。
- ○数字を先頭文字とする単語が扱える。
- ○カナ文字の単語が扱える。
- ○データのMSBを0にすることができ、これによってWORD STARのドキュメントファイルも扱える。
- ○10種の任意の記号を英字と同様に扱える。
- ○プリンター出力用のフォーマットで出力することができる。（CP/M版はプリンターへの出力のみ可。）

大好評 待望のFM-7用ジョイスティック

新発売￥12,800

※オートリピートスイッチ付（ギャラクシヤン等のソフトでジョイスティックと画面の動きとが連動します）テンキー対応（2, 4, 6, 8と1, 3, 5, の切換式）スペースキーと、ストップキーはpushボタン式



FM-11 スーパーぷりんていんぐユーティリティ

FM-11のユーザーへ お待たせしました!!

● 5インチDISK版 ￥12,800

ついにあなたのプリンタで（ドットインパクトタイプ）漢字印字が実現します。

FM CP/M86のPRINTINGがさらにパワーアップ!! その上F-BASIC(86)で漢字をふくんだLISTの印字もできます。

なお、印字の際に行間・横倍角・文字間隔など指定も可能!!

RUN85で動作する CP/M80のソフトも印字できます。

漢字プリンターを もっていない方の必需品

このソフトがあれば、あなたのプリンターも漢字プリンターに なってしまう。ものすごいソフト……

● 使用可能なプリンター

富士通シリアルドットプリンター・EPSON MPシリーズ RP-80・FP-80・FP-80K・KP-3000 インフォライナーRITEMAN・GP-550E（一部ソフトウェアの修正が必要です。）

★全国有名マイコンショップにてお求め下さい。

尚、通信販売も受付けております。送料500円

御振込先：住友銀行広島支店 当座201250

ミニファクス TEL (082)246-4592

UE URBAN アーバン電子株式会社

〒730 広島市中区大手町1-1-23 ☎(082)246-0993(代)

●プログラムを見やすく

REM文おもしろテクニック

佐藤 進一

市販されているソフトの中には，リストをとってみるとREM文に色がついていたりするものがあります。実はこれは，コントロールコード（ASCII コードの＄00から＄1Fまでを占領している得体の知れないもの）の応用なのです。単にREM文が色分けしているだけでも，ずいぶんプログラムが見やすくなると思いませんか？　そこで今回は，REM文の風変りな使い方と，代表的なコントロールコードの使い方について説明したいと思います。

D1 CHR$(&H11)

REM文に色をつけているものの正体は，実はこれなのです。ためしに，

```
10 PRINT CHR$(&H11)+"4 SAMPLE1"+" SAMPLE2"
```

を実行してみてください。COLOR文で色を変えたわけでもないのに，文字に色がついたでしょう？　次に，

```
20 PRINT CHR$(&H11)+"4 SAMPLE1";" SAMPLE2"
```

を加えて実行してみてください。20行の方では〝SAMPLE 2〟の色が変わらなかったでしょう。

▼図1

b7	b6	b5	b4	b3	b2	b1	b0
F	M	B	P	R	CL		

MSB　　　LSB

CL……文字の表示色を示します
R……R＝1のとき，文字の色を反転表示します
P……P＝1のとき，表示されている文字をユーザーの手で変更することができなくなります（変更しようとするとBEEP音がします）
B……未使用
M……M＝1のとき，ユーザーの手で変更されたことを示します
F……F＝1のとき，フィールドの先頭であることを示します。

実は，CHR＄(&H11) というのは，フィールドの始まりを定義するコントロールコードで，D1コードといってその後ろにはカラー情報などのアトリビュート文字が続きます。CHR＄(&H11) で定義されたアトリビュートの有効範囲は，フィールドの終わりか，次のアトリビュート文字があるところまでです。20行目では表示する文字列をつなぐのに〝;〟(セミコロン)を使っています。表示される文字列に違いはないのですが，フィールドはそこで途切れてしまうので〝SAMPLE2〟の方は，色が変わらなかった，というわけです。

```
30 PRINT CHR$(&H11)+"4 SAMPLE1"+CHR$(&H11)+
   "5 SAMPLE2"
```

を実行してみてください。今度は，2つとも色が変わるはずです。上記のリストではD1コードの直後にある〝4〟，〝5〟がアトリビュート文字になるわけです。

上のリストの場合，アトリビュート文字は〝4〟なのでASCIIコードは&H34，2進数に直すと，00110100となります。**図1**で確認してください。

D2 CHR$(&H12)

```
10 PRINT CHR$(&H12)+CHR$
   (20)+CHR$(10)+"SAMPLE"
```

まず，上のリストを実行してみてください。どうです？　どんな機能をもったコントロールコードか，すぐにおわかりでしょう。

これは，LOCATE文と同じなのです。文字列の区切りには，D1のときと同じように〝;〟や〝,〟でなく〝+〟を使います。D1が1バイトのデータを必要としたように，D2は2バイトのデータ（X座標とY座標）が必要です。ただしこの場合，座標は数値のままではなく，

CHR＄（座標値）

の形であたえてやらなければなりません。上のリストで座標をいろいろ変えて，ためしてみてください。

D3 CHR$(&H13)

これはBASICのSTRING＄と同じです。2バイトのデータが必要で，D3＋文字数＋繰り返す文字の形であたえます。次のリストを実行してみてください。

```
10 PRINT CHR$(&H13)+CHR$
   (10)+"A"
```

→ CHR$(&H1C)
← CHR$(&H1D)
↑ CHR$(&H1E)
↓ CHR$(&H1F)

これらは知っている人も多いと思います

が，カーソルを移動させるコントロールコードです。これについても，次のリストを実行して，動作を確認してください。

```
10 FOR␣I=1␣TO␣10
20 PRINT␣CHR$(&H1E)+CHR
   $(&H1E)+"SAMPLE"
30 NEXT␣I
```

BL CHR$(&H07)

これはBASICのBEEPと同じです。

```
PRINT␣CHR$(&H07)
```

を実行すると，BEEP音が出ます。また，コントロールコードを実行しても，カーソルの位置はかわりません。

LF CHR$(&H0A)

このコントロールコードは，カーソルを1行下げるもので，見かけの機能はCHR$(&H1F)の↓と同じですが，実はこれには重要な働きがあり，次に説明するCHR$(&H0D)——CR——とともに「改行コード」と呼ばれるものを構成しているのです。改行コード(CR, LF)というものは画面表示やプリンタへのデータの区切りや，ASCIIファイルのデータの区切りの信号として使われるもので，リターンキーを押したときに行われる動作も，実はこのCR，LFによるものなのです。

CR CHR$(&H0D)

このコントロールコードを実行すると，カーソルは今ある所の左端に移動します。同じ行に何回も違うデータを表示したいときなどに使うと便利です。

```
10 FOR␣I=1␣TO␣100：PRINT
   ␣I;CHR$(&H0D);：NEXT␣I
```

HM CHR$(&H0B)

これを実行すると，カーソルはホームポジション(座標(0,0)のところ)に移動します。

CL CHR$(&H0C)

画面をクリアするコントロールコードです。画面を消した後，カーソルはホームポジションに移ります。

コントロールコードのPRINT文への応用

これまで説明してきた数々のコントロールコードは，BASICのプログラム中でも結構便利な使い方ができるものです。例を挙げて説明しましょう。

```
10 CLS
20 LO$=CHR$(&H12)
30 DEF␣FNL$(X,Y,A$)=LO$
   +CHR$(X)+CHR$(Y)+A$
40 X=RND*40：Y=RND*20
50 PRINT␣FNL$(X,Y,"SAMPLE")
60 GOTO␣40
```

まず上のリストを実行してみてください。任意の座標に文字列などを表示したい場合，絶対にLOCATE文を使わなければならない，というようなことはないとおわかりいただけたと思います。次はCOLOR文も省く方法です。

```
10 CLS
20 LO$=CHR$(&H12)：COL$
   =CHR$(&H11)
30 DEF␣FNL$(X,Y,A$,C)=LO
   $+CHR$(X)+CHR$(Y)+
   COL$+CHR$(C)+A$
40 X=RND*40：Y=RND*20：C
   =RND*8
50 PRINT␣FNL$(X,Y,"SAMPLE",
   C)
60 GOTO␣40
```

と，前のリストをちょっと直して実行してみてください。また，別の使い方として，

```
10 A$=CHR$(&H12)+CHR$(15)
   +CHR$(10)+CHR$(&H11)
   +"6***␣"+CHR$(&H11)
   +"4SAMPLE"+CHR$(&H
   11)+"6␣***"
20 PRINT␣A$
```

のように，コントロールコードを文字変数の中に入れてやっても正しく機能するので，常に同じ座標に同じ文字列を表示したいときなどは，非常に便利です。またD1を使って色をつけた場合，実行し終わっても，COLOR文で設定した色は変わりませんのでプログラムのデバッグ時などはありがたいものです。

この他にもまだまだ応用の仕方はいろいろありますが，あとは**リスト3**のサンプルプログラムを参考にしてください。

コントロールコードのREM文への応用

これらのコントロールコードは，PRINT文中だけでなく，REM文の中においても正しく機能します。しかしコントロールコードは，直接キーインすることはできません。どうしたらよいのでしょうか。

ひとまず，次の手順を実行してみてください。

```
10 '#4SAMPLE
```

まず，REM文を打ち込み，次にその行のリストをとります。

```
LIST 10↵
10 '#4 SAMPLE
```

次にモニタを呼び出し，&H00A2番地からの内容を見ます。

```
MON↵
*DA2
00A2  07 90……
00AA  ……………
```

&H00A2, &H00A3番地は，最後にリストをとった行のBASICのテキストが納められている番地を示しているので，次にその番地の内容を見てみます。

```
*D0790↵
0790 07 9F 00 0A 3A 8D 23 34
0798 53 41 4D 50 4C 45 00 00
```

先頭の079Fは次のテキストが格納される番地，次の000A（10進数では10）は行番号，3A8DというのがREM文の中間コードです。このあとから00のところまでが，REM文の文字列が格納されている場所であるというわけです。そして，REM文の先頭に置いた"#"(ASCIIコードで23)をD1に書き換えてみます。

```
*M0796↵
0796 23—11↵
0797 34—. ↵
```

以上の作業が終わったら，BREAKキーを押してBASICに戻り，リストをとってみてください。REM文に色がついたでしょう。

しかし，REM文１個作るのにいちいちこんな手間をかけるのは大変なので，自動的にこの変換作業をするプログラムを作りました。そのアセンブルリストを**リスト１**に，ダンプリストを**リスト２**に示しました。

このプログラムはリロケータブルなので，どこのアドレスに置いても動きます。

EXEC　先頭アドレス

で変換を行い，

EXEC　先頭アドレス＋２

で逆変換を行います。なお，変換される文字は**図２**のとおりです。皆さんの手持ちのプログラムに次のリストを入れて確かめてください。

```
××××  '#3 SAMPLE1 #5 SA
       MPLE2
××××  '@#6 SAMPLE3
××××  '日#2 SAMPLE4
××××  '年#1 SAMPLE5
××××  '時 SAMPLE6
```

（××××：適当な行番号）

▼図２

#	↔	D1
$	↔	D2
%	↔	D3
@	↔	CR
円	↔	LF
年	↔	BL
月	↔	HM
日	↔	CL
時	↔	←
分	↔	↑

リスト１　REM文変換プログラムアセンブルリスト

©1983 S. SATOH

```
PAGE  001 (821201,012823) REMARK

                                NAM    REMARK     ; ←BASICのテキストの先頭アドレスのポインタ
                                OPT    SYM,MEM    ; ←BASICのテキストの終了アドレスのポインタ
                         *
                0033     TXTTOP EQU    $0033
                0035     TXTEND EQU    $0035
                         *
      6800                      ORG    $6800
                         *
      6800 20   02   6804 ENTRY BRA    REMREM  REM -> Spl.REM
      6802 20   0C   6810       BRA    DISREM  Spl.REM -> REM
      6804 34   7F       REMREM PSHS   D,X,Y,U,DP,CC
      6806 31   8D 003A         LEAY   DATA1,PCR  ; Y:被変換キャラクタテーブル先頭
      680A 33   8D 0040         LEAU   DATA2,PCR  ; U:変換キャラクタテーブル先頭
      680E 20   0A   681A       BRA    START      ; REM文→特殊REM文のためのデータをセット
                         *
      6810 34   7F       DISREM PSHS   D,X,Y,U,DP,CC
      6812 31   8D 0038         LEAY   DATA2,PCR  ; Y:被変換キャラクタテーブル先頭
      6816 33   8D 002A         LEAU   DATA1,PCR  ; U:変換キャラクタテーブル先頭
                                                  ; 特殊REM文→REM文のためのデータをセット
                         *
      681A 9E   33       START  LDX    TXTTOP     ; メインプログラム
      681C EC   80       LOOP   LDD    ,X+
      681E 1083 3A8D            CMPD   #$3A8D     ; REM文？
      6822 27   07   682B       BEQ    CHECK
      6824 9C   35       RETN   CMPX   TXTEND
      6826 25   F4   681C       BLO    LOOP
      6828 35   7F              PULS   D,X,Y,U,DP,CC
      682A 39                   RTS
                         *
      682B A6   84       CHECK  LDA    ,X         ; REM文終わり？
      682D 27   F5   6824       BEQ    RETN
      682F C6   00              LDB    #0
      6831 A1   A5       LOOP2  CMPA   B,Y        ; 被変換キャラクタに該当？
      6833 27   09   683E       BEQ    OUT        ; （10個チェックして該当しないなら変換なし）
      6835 5C                   INCB
      6836 C1   0A              CMPB   #10
      6838 26   F7   6831       BNE    LOOP2
      683A 30   01              LEAX   1,X
      683C 20   ED   682B       BRA    CHECK
      683E A6   C5       OUT    LDA    B,U        ; キャラクタの置き換え
      6840 A7   80              STA    ,X+
      6842 20   E7   682B       BRA    CHECK
```

```
01450                        *
01460  6844      23          DATA1  FCB    $23,$24,$25,$40,$F1,$F2,$F3,$F4,$F5,$F6
                                           #   $   %   @   円  年  月  日  時  分
       6845      24
       6846      25
       6847      40
       6848      F1
       6849      F2
       684A      F3
       684B      F4
       684C      F5
       684D      F6
01470  684E      11          DATA2  FCB    $11,$12,$13,$0D,$0A,$07,$0B,$0C,$1D,$1E
                                           D1  D2  D3  CR  LF  BL  HM  CL  ←   ↑
       684F      12
       6850      13
       6851      0D
       6852      0A
       6853      07
```

```
PAGE  002 (821201,012823) REMARK

       6854      0B
       6855      0C
       6856      1D
       6857      1E
01480                        *
01520            6800               END    ENTRY
TOTAL ERRORS 00000--00000
TOTAL WARNINGS 00000--00000

PROGRAM BEGIN ADDR=6800
PROGRAM END   ADDR=6857
PROGRAM ENTRY ADDR=6800


CHECK  682B  DATA1  6844  DATA2  684E  DISREM 6810  ENTRY  6800  LOOP   681C
LOOP2  6831  OUT    683E  REMREM 6804  RETN   6824  START  681A  TXTEND 0035
TXTTOP 0033
```

リスト2 REM文変換プログラムダンプリスト

```
Add   +0 +1 +2 +3 +4 +5 +6 +7  Sum
6800  20 02 20 0C 34 7F 31 8D :BF
6808  00 3A 33 8D 00 40 20 0A :64
6810  34 7F 31 8D 00 38 33 8D :69
6818  00 2A 9E 33 EC 80 10 83 :FA
6820  3A 8D 27 07 9C 35 25 F4 :DF
6828  35 7F 39 A6 84 27 F5 C6 :F9
6830  00 A1 A5 27 09 5C C1 0A :9D
6838  26 F7 30 01 20 ED A6 C5 :C6
6840  A7 80 20 E7 23 24 25 40 :DA
6848  F1 F2 F3 F4 F5 F6 11 12 :D8
6850  13 0D 0A 07 0B 0C 1D 1E :83
```

おまけ

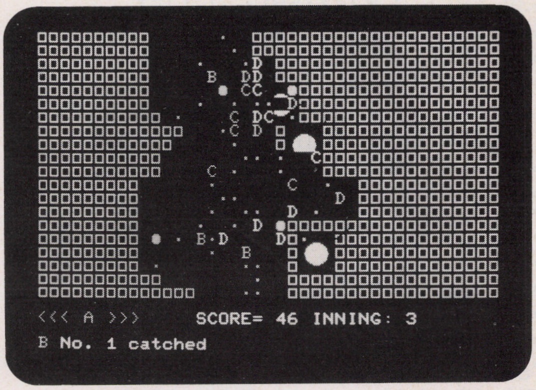
リスト3にコントロールコードなどの応用例として作ったサンプルプログラムを挙げます。ちょっとしたシミュレーションゲームで，２人用の「スタートレック」みたいなものです。**リスト３**を正しく打ち込んでRUNするとタイトルが表示された後，各プレイヤーの名前を尋ねてきますので先攻，後攻を決めて，先攻の人はPLAYER１,後攻の人はPLAYER２にそれぞれ入力してください。ゲームが始まるとマップが表示され，各プレイヤーの艦隊と各艦のレーダーの範囲内にある物を表示します。このとき画面上のキャラクタはそれぞれ，〝B″が戦艦(Battleship)，〝C″が巡洋艦(Cruiser)，〝D″が駆逐艦 (Destroyer) です。なお，〝・″，〝●″は隕石や小惑星で，船の移動のさまたげになります。

マップが表示されたら各プレイヤーは自分の船を移動させます。０キーで停止，１～９キーで移動，Wキーでワープできます。なお戦艦，巡洋艦は２コマ，駆逐艦は３コマ動けます。ワープのときは目標地点のX，Y座標を現在いる場所から数えた相対座標で入れます。ただし，距離が遠いほど，ワープ終了地点の誤差も大きくなります。

18隻全部の移動が終わると，攻撃にうつります。有効射程はその船のレーダーに見えているところまでで，攻撃目標はコンピュータが自動的に捜します。ただ，一度にあまり多くの目標をとらえると，攻撃力が有効に生かされないので要注意です。

このようにして各プレイヤーが，交互に自分の艦隊を動かしていき，相手の艦隊を全滅させた方の勝ちです。隕石や小惑星がごろごろしていますので，よく地の利を考えて作戦をたててください。ではご健闘を祈ります。

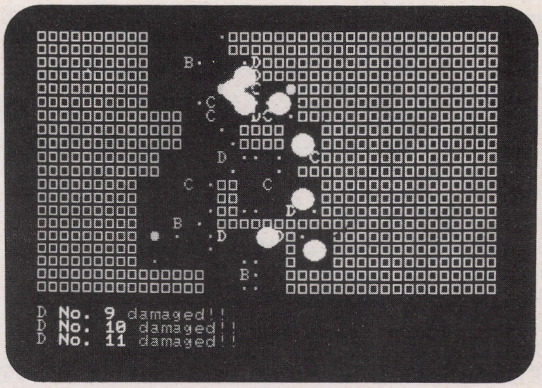
リスト３




```
1000 '@#0******
1010 '@#4<<< MAIN >>>
1020 '@#0******
1030 CLEAR2000
1040 GOSUB1120:'#1INITIALIZE
1050 ING=1
1060 FOR I=0 TO 1
1070 I2=(I+1)MOD2
1080 GOSUB1940:'#2CHECK
1090 GOSUB1440:'#3MOVE SHIPS
1100 GOSUB1720:'#4ATTACK
1110 NEXT I:ING=ING+1:GOTO1060
1120 '@#0******
1130 '@#1<<< INITIALIZE >>>
1140 '@#0******
1150 DEFINT A-Z
1160 RANDOMIZE TIME MOD1000
1170 COLOR7,0:WIDTH40,25:CONSOLE21,4,0,0:GOSUB2970:'#4TYTLE
1180 DIM MAP$(39,19),SH$(1,17),SH(1,17,2),TG(17,18),AT(17)
1190 FOR Y=0 TO 19:FOR X=0 TO 39:MAP$(X,Y)=" ":NEXT X,Y
1200 CO$=CHR$(17)
1210 LO$=CHR$(18)
1220 DEF FNR(X)=INT(RND*X)
1230 DEF FNS$(I,J)=LO$+CHR$(SH(I,J,0))+CHR$(SH(I,J,1))+SH$(I,J)
1240 DEF FNM$(X,Y)=LO$+CHR$(X)+CHR$(Y)+MAP$(X,Y)
1250 DEF FNC$(X,Y)=LO$+CHR$(X)+CHR$(Y)
1260 CL1$=FNC$(0,23)+CHR$(19)+CHR$(38)+" "+CHR$(13)
1270 FOR I=0 TO 2:SH$(0,I)=CO$+"4B":SH$(1,I)=CO$+"3B":SH(0,I,0)=FNR(10)+5:SH(0,I
,1)=FNR(20):SH(0,I,2)=2500+FNR(1000):SH(1,I,0)=FNR(10)+25:SH(1,I,1)=FNR(20):SH(1
,I,2)=2500+FNR(1000):NEXT
1280 FOR I=3 TO 7:SH$(0,I)=CO$+"4C":SH$(1,I)=CO$+"3C":SH(0,I,0)=FNR(10)+8:SH(0,I
,1)=FNR(20):SH(0,I,2)=1500+FNR(1000):SH(1,I,0)=FNR(10)+22:SH(1,I,1)=FNR(20):SH(1
,I,2)=1500+FNR(1000):NEXT
1290 FOR I=8 TO 17:SH$(0,I)=CO$+"4D":SH$(1,I)=CO$+"3D":SH(0,I,0)=FNR(10)+10:SH(0
,I,1)=FNR(20):SH(0,I,2)=500+FNR(1000):SH(1,I,0)=FNR(10)+20:SH(1,I,1)=FNR(20):SH(
1,I,2)=500+FNR(1000):NEXT
1300 AST1$=CO$+"2·":AST2$=CO$+"2●":AST3$=CO$+"1·"
1310 FOR I=0 TO FNR(100):X=FNR(40):Y=FNR(20):MAP$(X,Y)=AST1$:NEXT
1320 FOR I=0 TO FNR(50):X=15+FNR(10):Y=FNR(20):MAP$(X,Y)=AST1$:NEXT
1330 FOR I=0 TO FNR(30):X=FNR(40):Y=FNR(20):MAP$(X,Y)=AST2$:NEXT
1340 FOR I=0 TO FNR(30):X=FNR(40):Y=FNR(20):MAP$(X,Y)=AST3$:NEXT
1350 FOR I=0 TO 1:FOR J=0 TO 17:MAP$(SH(I,J,0),SH(I,J,1))=SH$(I,J):NEXT J,I
1360 X(1)=-1:X(3)=1:X(4)=-1:X(6)=1:X(7)=-1:X(9)=1
1370 Y(1)=1:Y(2)=1:Y(3)=1:Y(7)=-1:Y(8)=-1:Y(9)=-1
```

```
1380 INPUT"NAME OF PLAYER 1";NAM$(0)
1390 INPUT"NAME OF PLAYER 2";NAM$(1)
1400 NA$(0)=FNC$(0,21)+CO$+"4<<< "+NAM$(0)+" >>>"
1410 NA$(1)=FNC$(0,21)+CO$+"3OOO "+NAM$(1)+" OOO"
1420 SCORE(0)=0:SCORE(1)=0
1430 RETURN
1440 '@#0******
1450 '@#3<<< MOVE SHIPS >>>
1460 '@#0******
1470 IF STN<>0 THEN GOSUB2470:'#6PRINT MAP
1480 FOR J=0 TO 17
1490 IF SH(I,J,2)=0 THEN 1700
1500 NN=2:IF J>7 THEN NN=3
1510 PRINTCL1$+SH$(I,J);" No.";J;"ENERGY";SH(I,J,2);
1520 FOR N=1 TO NN
1530 X=SH(I,J,0)*16:Y=SH(I,J,1)*8
1540 SYMBOL(X,Y),"■",2,1,7,,XOR
1550 DIR$=INPUT$(1)
1560 IF DIR$="W" OR DIR$="w" THEN GOTO2560:'#3WARP
1570 IF DIR$="0" OR DIR$=" " THEN PRINTFNS$(I,J);:GOTO1690
1580 DIR=VAL(DIR$)
1590 X=SH(I,J,0)+X(DIR):Y=SH(I,J,1)+Y(DIR)
1600 IF X<0 OR X>39 OR Y<0 OR Y>19 THEN 1550
1610 IF MAP$(X,Y)<>" " THEN 1550
1620 MAP$(SH(I,J,0),SH(I,J,1))=" ":PRINTFNC$(SH(I,J,0),SH(I,J,1))+" ";
1630 SH(I,J,0)=X:SH(I,J,1)=Y
1640 MAP$(X,Y)=SH$(I,J)
1650 BEEP1:BEEP0
1660 ST=SH(I,J,2)/1000
1670 FOR Y2=Y-ST TO Y+ST:FOR X2=X-ST TO X+ST:IF X2<0 OR X2>39 OR Y2<0 OR Y2>19 T
HEN 1680 ELSE PRINTFNM$(X2,Y2);
1680 NEXT X2,Y2
1690 NEXT N
1700 NEXT J
1710 RETURN
1720 '@#0******
1730 '@#4<<< ATTACK >>>
1740 '@#0******
1750 STN=0:FOR J=0 TO 17
1760 TG(J,0)=0
1770 IF SH(I,J,2)=0 THEN 1850
1780 PRINTCL1$+SH$(I,J);" No.";J;"catched";
1790 X=SH(I,J,0):Y=SH(I,J,1):ST=SH(I,J,2)/1000:TN=0
1800 FOR Y2=Y-ST TO Y+ST:FOR X2=X-ST TO X+ST
1810 IF X2<0 OR X2>39 OR Y2<0 OR Y2>19 THEN 1820 ELSE GOSUB1870
1820 NEXT X2,Y2
1830 TG(J,0)=TN
1840 IF TN=0 THEN PRINT" nothing...";:BEEP1:BEEP0
1850 NEXT J
1860 RETURN
1870 '#4<<< TARGET >>>
1880 C=VAL(MID$(MAP$(X2,Y2),2,1))
1890 IF C=0 OR C=2 OR C=1 OR C=4-I THEN RETURN
1900 FOR J2=0 TO 17
1910 X3=SH(I2,J2,0):Y3=SH(I2,J2,1)
1920 IF X3=X2 AND Y3=Y2 AND SH(I2,J2,2)>0 THEN TN=TN+1:STN=STN+1:TG(J,TN)=J2:PRI
NT" ";SH$(I2,J2);:BEEP:BEEP
1930 NEXT J2:RETURN
1940 '@#0******
1950 '@#2<<< CHECK >>>
1960 '@#0******
1970 GOSUB2430:'#6PRINT MAP
1980 IF STN=0 THEN RETURN
1990 FOR K=0 TO 17:AT(K)=0:NEXT K
2000 FOR J2=0 TO 17
2010 IF TG(J2,0)=0 THEN 2100
2020 X1=SH(I2,J2,0)*16+8:Y1=SH(I2,J2,1)*8+4
2030 FOR K=1 TO TG(J2,0)
2040 N=TG(J2,K)
```

```
2050 X2=SH(I,N,0)*16+8:Y2=SH(I,N,1)*8+4
2060 LINE(X1,Y1)-(X2,Y2),PSET,6
2070 GOSUB2940:'#6SOUND(BEAM)
2080 AT(N)=AT(N)+SH(I2,J2,2)/TG(J2,0)
2090 NEXT K
2100 NEXT J2
2110 FOR J2=0 TO 17
2120 X1=SH(I2,J2,0)*16+8:Y1=SH(I2,J2,1)*8+4
2130 IF TG(J2,0)=0 THEN 2190
2140 FOR K=1 TO TG(J2,0)
2150 N=TG(J2,K)
2160 X2=SH(I,N,0)*16+8:Y2=SH(I,N,1)*8+4
2170 LINE(X1,Y1)-(X2,Y2),PRESET
2180 NEXT K
2190 NEXT J2
2200 SOUND 6,0:SOUND 8,8:SOUND 7,247
2210 FOR R=0 TO 15
2220 FOR J=0 TO 17
2230 IF AT(J)=0 THEN 2270
2240 X2=SH(I,J,0)*16+8:Y2=SH(I,J,1)*8+4
2250 CIRCLE(X2,Y2),R,7
2260 SOUND 6,R*2+1
2270 NEXT J,R
2280 SOUND 6,0:SOUND 7,255
2290 LOCATE0,24
2300 FOR J=0 TO 17
2310 IF AT(J)=0 THEN 2380
2320 DME=(SH(I,J,2)-AT(J))*(1+RND)/2
2330 IF DME>=0 THEN 2380
2340 SCORE(I2)=SCORE(I2)-DME/100
2350 SH(I,J,2)=SH(I,J,2)+DME
2360 IF SH(I,J,2)<=0 THEN SH(I,J,2)=0:MAP$(SH(I,J,0),SH(I,J,1))=" ":PRINTSH$(I,J
);" No.";J;CO$+"2sinked!!!":BEEP:GOSUB2410:GOTO2380
2370 PRINTSH$(I,J);" No.";J;CO$+"2damaged!!":BEEP:GOSUB2410
2380 NEXT J
2390 SUM1=0:FOR J=0 TO 17:SUM1=SUM1+SH(I,J,2):NEXT J:IF SUM1<=0 THEN 3070:'GAME
OVER
2400 RETURN
2410 '#4<<< DELAY >>>
2420 FOR H=1 TO 500:NEXT H:RETURN
2430 '@#0******
2440 '@#6<<< PRINT MAP >>>
2450 '@#0******
2460 LINE(0,0)-(39,19),"□",1,BF
2470 CLS1:PRINTNA$(I),"SCORE=";SCORE(I);"INNING:";ING;:FOR J=0 TO 17
2480 IF SH(I,J,2)=0 THEN 2540
2490 ST=SH(I,J,2)/1000
2500 X=SH(I,J,0):Y=SH(I,J,1)
2510 FOR Y2=Y-ST TO Y+ST:FOR X2=X-ST TO X+ST
2520 IF X2<0 OR X2>39 OR Y2<0 OR Y2>19 THEN 2530 ELSE PRINTFNM$(X2,Y2);
2530 NEXT X2,Y2
2540 NEXT J
2550 RETURN
2560 '@#0******
2570 '@#3<<< WARP >>>
2580 '@#0******
2590 PRINTCL1$;:INPUT" X=",X:PRINTCL1$;:INPUT" Y=",Y
2600 PRINTFNC$(0,23)+CO$+"2*** WARP!! ***      ":GOSUB2960:'#3SOUND(WARP)
2610 X2=SH(I,J,0)*16+8:Y2=SH(I,J,1)*8+4
2620 FOR R=4 TO 36 STEP2
2630 R2=R:IF R>32 THEN R2=32
2640 CIRCLE(X2,Y2),R2,6
2650 CIRCLE(X2,Y2),R-4,0
2660 NEXT R
2670 X2=SH(I,J,0):Y2=SH(I,J,1):MAP$(X2,Y2)=" ":ST=SH(I,J,2)/1000
2680 FOR Y3=Y2-ST TO Y2+ST:FOR X3=X2-ST TO X2+ST
2690 IF X3<0 OR X3>39 OR Y3<0 OR Y3>19 THEN 2700 ELSE PRINTFNM$(X3,Y3);
2700 NEXT X3,Y3
```



```
2710 R=SQR(X*X+Y*Y)
2720 X2=X2+X+FNR(R/2)-(R/4):Y2=Y2+Y+FNR(R/2)-(R/4)
2730 IF X2<0 OR X2>39 OR Y2<0 OR Y2>19 THEN PRINTCL1$+CO$+"2Run out to the outer
-space!!";:SH(I,J,2)=0:BEEP:GOSUB2410:GOTO1700:'RETURN
2740 IF MAP$(X2,Y2)<>" " THEN PRINTCL1$+CO$+"2Clashed with anyting!! Can not war
p-ont!!";:SH(I,J,2)=0:BEEP:GOSUB2410:GOTO1700:'RETURN
2750 SH(I,J,2)=SH(I,J,2)-R*20
2760 IF SH(I,J,2)<=0 THEN PRINTCL1$+CO$+"2Energy-out!! Can not warp-out!!":SH(I,
J,2)=0:BEEP:GOSUB2410:GOTO1700:'RETURN
2770 SH(I,J,0)=X2:SH(I,J,1)=Y2
2780 X2=X2*16+8:Y2=Y2*8+4
2790 FOR R=4 TO 36 STEP2
2800 R2=R:IF R>32 THEN R2=32
2810 CIRCLE(X2,Y2),R2,6
2820 CIRCLE(X2,Y2),R-4,0
2830 NEXT R
2840 X2=SH(I,J,0):Y2=SH(I,J,1)
2850 MAP$(X2,Y2)=SH$(I,J):ST=SH(I,J,2)/1000
2860 FOR Y3=Y2-ST TO Y2+ST:FOR X3=X2-ST TO X2+ST
2870 IF X3<0 OR X3>39 OR Y3<0 OR Y3>19 THEN 2880 ELSE PRINTFNM$(X3,Y3);
2880 NEXT X3,Y3
2890 GOTO1690:'RETURN
2900 '@#0******
2910 '@#4<<< SOUNDS >>>
2920 '@#0******
2930 '#6<<< BEAM >>>
2940 SOUND 0,0:SOUND 1,0:SOUND 8,8:SOUND 7,254:FOR A=1 TO 10:FOR B=80 TO 40 STEP
-2:SOUND 0,B:NEXT B,A:SOUND 0,0:SOUND 7,255:RETURN
2950 '#3<<< WARP >>>
2960 SOUND 0,0:SOUND 1,0:SOUND 8,8:SOUND 7,254:FOR A=254 TO 0 STEP-8:FOR B=0 TO
2:SOUND 0,A/2^B:NEXT B,A:SOUND 0,0:SOUND 7,255:RETURN
2970 '@#0******
2980 '@#4<<< TYTLE >>>
2990 '@#0******
3000 T$="ASTEROID FLEET"
3010 SYMBOL(36,62),T$,5,2,5
3020 SYMBOL(44,62),T$,5,2,5
3030 SYMBOL(44,66),T$,5,2,5
3040 SYMBOL(36,66),T$,5,2,5
3050 SYMBOL(40,64),T$,5,2,1
3060 RETURN
3070 '@#0******
3080 '@#2<<< GAME OVER >>>
3090 '@#0******
3100 WIDTH40,25
3110 SYMBOL(40,64),"GAME OVER",8,2,2
3120 SYMBOL(42,66),"GAME OVER",8,2,2
3130 PRINTFNC$(5,15)+CO$+"4"+NAM$(0),SCORE(0);"POINT"
3140 PRINTFNC$(5,16)+CO$+"3"+NAM$(1),SCORE(1);"POINT"
3150 PRINT:PRINT
3160 IF SCORE(0)=SCORE(1) THEN PRINT"    *** DRAWN GAME ***":GOTO 3200
3170 IF SCORE(0)>SCORE(1) THEN I=0:I2=1 ELSE I=1:I2=0
3180 IF SCORE(I)>SCORE(I2)*10 THEN PRINT"   *** TOTAL VICTORY FOR ";NAM$(I);" **
*":GOTO3200
3190 IF SCORE(I)>SCORE(I2)*3 THEN PRINT"   *** MAJOR VICTORY FOR ";NAM$(I);" ***
" ELSE PRINT"   *** MINOR VICTORY FOR ";NAM$(I);" ***"
3200 PRINT:PRINT:PRINTCO$+"2   Try again ? (Y/N)"
3210 A$=INPUT$(1):IF A$="Y" OR A$="y" THEN RUN
3220 IF A$="N" OR A$="n" THEN END ELSE 3210
```

FM-7(,8)による

6809機械語の書き方 〔前〕

草津　酔床

BASICでもそうですが，機械語を紙とエンピツだけでマスターするのはたいへんです。なるべく実験を中心にして，目で確かめながら勉強することにしましょう。

なお，ディスクをお持ちの方は，ディスクのスイッチを切って，ROM BASICにしてください。

1 16進数

機械語をやるからには，どうしても16進数を避けて通れません。いやなことは最初にかたづけてしまうことにしましょう。

小学校のときに，「10進数以外の数」として，2進数や5進数の勉強をさせられましたね。たとえば2進数では数字は0と1の2つだけだし，5進数では0から4まで5個の数字を使うのでした。

16進数でも同じことで，16個の数字が使われます。しかし，ふつうの数字は0から9までの10個しかないので，そのほかにAからFまでを数字として使うことにしています。Aは10進数に直せば10，Bが11です（表1）。

さっそくマイコンを使ってみましょうか。表2のプログラムをRUNさせてください。これで16進数を10進数に変換させることができます。たとえば「A」を入力すると，答は10だし，「F」ならば15です。

Fの次は，やっと2桁になって10です。つまり16進数の10は10進数の16ですね。20はその2倍だから，10進数では32というわけです。

「100」と入れれば，答の256が表示されます。10進数の100は，もちろん10×10ですが，16進数の100は16×16というわけですね。それより1つ少ない255は16進数では「FF」で，これが2桁で最大の数ということになります。

2 CPUとメモリ

16進数の勉強を終わったところで，FM-7の中を調べてみましょう。

まずメモリですが，これには16進数で4桁の番号がつけられています。各番地のサイズは16進数で2桁ですから，それぞれ00からFFまでの値をとることになります。

メモリの番地はいくつあるでしょうか。16進数の4桁ですから，0000番地からFFFF番地まで，すなわち16進数で10000ですね。10進数に直せば16の4乗ですが，さっきの表2のプログラムを使って10000を入れてやると，答は65536です。インタフェース誌によれば，技術屋さんは「ムコのゴンザブロウ」と覚えるのだそうです。

次はいよいよCPUの6809ですが，その主役は「アキュムレータ」といわれる計算専門の記憶装置で，AとBの2つがあります。AもBも16進数で2桁ですから，サイズはメモリと同じですね。

このほかにXおよびYという名前の「インデックス」と呼ばれる記憶装置があります。AとBが計算の専門家であるのに対し，XとYは主としてメモリの番地を指定するのに使われます。番地指定用ですからサイズも16進数で4桁となっています（図1）。

3 USRの使い方

ご承知のとおり，BASICから機械語サブルーチンを呼ぶには，USRを使います。ちょっと練習してみましょう。

ブザーをONにするには，BASICではBEEP 1ですが，機械語はFM-7の場合，DB3F(FM-8はDD3E)番地から書かれています。またBEEP 0はDB44(同DD43)番地からです。

それでは表3のプログラムをRUNさせてみましょう。ブザーが鳴り続けましたね。ブレークキーを押せば止まります。同様に表4では，ブザーがピッと鳴ります。なお，プログラム中の&Hは，16進数を表わす記号です。

このように，機械語サブルーチンを呼ぶためのプログラムは，

(1) あらかじめ「DEF　USR×＝……」の形で，番地をCPUに教えておく。なお×は0から9まで。

(2) そのサブルーチンを呼ぶときは，「V＝USR×(V)とする。なお変数はVでなくとも，AでもBでもかまいません。

というようにします。

4 ROMとRAM

機械語モニタのDコマンドを使って，メモリの内容を見てみましょう。

MON⊘

と打ち込むと，＊が表示されます。これでモニタのレベルになりました。次にDコマンドでメモリの内容をのぞいてみましょう。

DDB3F⊘

と打ち込んでください。

さっきのDB3F番地

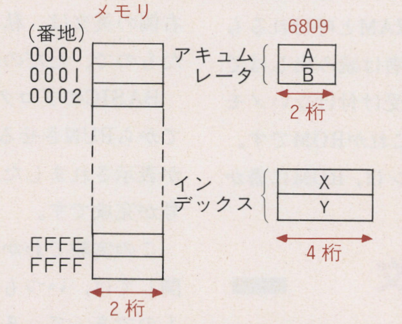
表1　16進数と10進数

16ｼﾝ	10ｼﾝ
0	0
1	1
....	
9	9
A	10
B	11
C	12
D	13
E	14
F	15

表2　16進・10進変換

```
200 D=0
210 INPUT H$
220 FOR I=1 TO LEN(H$)
230 V$=MID$(H$,I,1)
240 V=VAL("&H"+V$)
250 D=D*16+V
260 NEXT I
270 PRINT D
280 GOTO 200
```

図1　6809とメモリ



©1983 Y. KUSATSU

表3　BEEP 1

```
200 DEF USR0=&HDB3F
210 V=USR0(V)
220 END
```

表4　短いBEEP

```
200 DEF USR0=&HDB3F
210 DEF USR1=&HDB44
220 V=USR0(V)
230 FOR I=1 TO 20
240 NEXT I
250 V=USR1(V)
260 END
```

あたりは，びっしり16進数が書かれています。前述のとおりメモリの番地は0000からFFFF番地までですが，このうち8000以上の番地は，みんなさまざまの16進数で埋められていますね。ところがゼロ番地から7FFF番地までは，ほとんど「00」です。

また実験をやってみましょう。ゼロ番地から7FFF番地までの間で，なるべく「00」が並んでいるところを選んで，モニタのMコマンドを使って，適当な16進数をどんどん書き込んでみてください。次に再びDコマンドで調べてみると，もちろん読者の書き込んだとおりの数値が表示されます。

この実験のとき，もしも機械の動作がおかしくなったら，リセットスイッチを押してください。故障ではありません。 実は一部のメモリはCPUが自分の作業用に使っているのですが，それを人間が勝手に書きかえてしまったので，誤動作したわけです。

次に8000番地以上のメモリについて，同じ実験をしてみましょう。Mコマンドで好きな数を書き込んでからDコマンドで調べると，意外にも書いたつもりの16進数は書かれていなくて，データは元のままであることがわかります。

——この実験で，重要な発見ができました。0000から7FFF番地までと，8000番地以上とは，メモリの性質が違います。前者は内容を読むことはもちろん，書き込みも自由にできるメモリで，RAMと呼ばれるものです。これに対して後者は読み出しはできるけれども書き込みは受け付けないメモリが配置されています。これがROMです。BEEP関係のサブールチンは，ROMに書かれているわけですね。

5 CLEAR文

さっきのBEEP関係の機械語のように，メーカーの提供するサブルーチンをそのまま使っているうちはよいのですが，自分で機械語を書こうというときは，気をつけなければならない点があります。

さっきもちょっと出た話ですが，人間が機械語を書くときに，CPUが使っている作業用メモリを勝手に書きかえてしまうことがあります。逆にまたCPUの方も，特に人間から指示を受けない限り，すべてのメモリは自分が自由に使えると思っているわけですから，へたをすると，せっかく人間が書き込んだ機械語プログラムを，CPUが変更してしまうおそれがあります。

CPUが自由に使えるメモリの範囲を限定するには，CLEAR文を使います。たとえば

CLEAR300, &H6FFF

としておくと，CPUの使用範囲は6FFF番地までとなりますから，人間が機械語プログラムを書くのに，7000番地からRAMの終端の7FFF番地まで使うことができます。

6 初めて書く機械語

さしあたって使う6809用の命令表が**表5**です。このほかFM-7専用のキマリ文句として，**表6**も見てください。

まずD08E番地のキマリ文句を使って，ディスプレイに「5」と書かせてみましょう。

(1) 「5」のキャラクタコードは，マニュアルによれば16進で35ですから，アキュムレータAに35を入れます。

(2) D08E番地のサブルーチンにジャンプします。

(3) 以上で作業は終わりですから，リターンします。

機械語プログラムは7000番地から書くことにしましょう。**表7**のとおり，間違えないように「86　35 BD……」と，モニタのMコマンドを使って書き込みます。なお右側の英文は，私が適当にコメントをつけたもので，ただの参考です。

BASICのプログラムを**表8**のとおり書いてからRUNさせると，ディスプレイに「5」が表示されました。機械語プログラム第1号が完成です。

この実験でわかるとおり，テレビ画面に関しては，いつもキャラクタコードを使うものであって，5を表示するのだからといってA＝5にしてもダメです。ついでに実験してみてください。テレビ画面には何も表示されません。

もう1つ，キーボードからの入力もやってみましょう。7000番地からのプログラムを**表9**のように書きかえてRUNさせます。ここでたとえば5キーを押してから，Dコマンドで700F番地を調べると，そのキャラクタコードの35が入っています。

このプログラムを使うと，他のキーのキャラクタコードを知ることができます。RUNさせてからESCキーを押すと，700F番地は1Bになっています。すなわちESCキーのキャラクタコードは1Bであることがわかります。

表5　6809　主要な命令表

1　データを入れる命令

Aに××を	86　××
Bに××を	C6　××

2　コピー命令

○○××番地→A	B6　○○　××
A→○○××番地	B7　○○　××
○○××番地→B	F6　○○　××
B→○○××番地	F7　○○　××
A→B	1F　89
B→A	1F　98

3　計算命令(AおよびB)

	A	B
××を加えよ	8B　××	CB　××
××を引け	80　××	C0　××
××と比較せよ	81　××	C1　××
1を加えよ	4C	5C
1を引け	4A	5A

4　相対ジャンプ(ブランチ)命令

無条件	20　××
ゼロのとき	27　××
ゼロでないとき	26　××
キャリーのとき	25　××
キャリーでないとき	24　××

5　サブルーチン関係の命令

○○××番地のサブルーチンにジャンプせよ	BD ○○ ××
元に戻れ	39

表6　FM-7の入出力命令

命令	内　容	備　　考
D08E	テレビ画面に1字出力	あらかじめアキュムレータAにキャラクタコードを入れておく
D072	キーボードから1字入力	戻ったときアキュムレータAにキャラクタコードがはいっている

表7　「5」と表示

```
7000: 86 35       LET A=&H35
7002: BD D0 8E    PRINT 1 CHARACTER
7005: 39          RETURN
```

表8　BASICの主プログラム

```
200 CLEAR 300,&H6FFF
210 DEF USR0=&H7000
220 V=USR0(V)
230 END
```

表9　キー入力を700F番地にストア

```
7000: BD D0 72    INPUT 1 CHARACTER
7003: B7 70 0F    STORE Acc.A IN (&H700F)
7006: 39          RETURN
```

7 FM-8, FM-11への応用

FM-8, FM-11も同じ6809CPUを使っていますから，**表5**の命令はすべて共通です。ただキマリ文句は，機種が違うと変わりますから，**表6**の命令は使えません。

調べてみたところ，1字出力は，FM-8がD352番地，FM-11がDE22番地，1字入力は，FM-8がD336番地，FM-11がDE08番地でした。したがって，**表7**のマシン語データD08Eを，FM-8ユーザーはD352，FM-11ユーザーはDE22に，また**表9**および後述の**表14**のD072も同様にD336あるいはDE08に直してください。

また，BEEP関係のプログラム(**表3**, **表4**, **表11**)のDB3F，DB44も，FM-8ユーザーは，それぞれDD3E，DD43に変更します。

8 コーヒーブレーク

機械語はまったく初めてという読者には，新しい話ばかりで，先が思いやられると感じておられるでしょう。

ドイツのことわざで,「何でもはじめはきつい」といいますし，イギリスでも「ひとたび手をつければ，半分はできたようなものだ」とかいうのがありました。

とくに機械語の勉強にあたっては，いちばんたいへんなのは最初の入口のところで，先へ進むほど新しい手法はなくなってきます。そのかわり，しだいに単調な灰色の作業になっていくわけで，おそらく読者がこの連載を読み終わるころには，「難しい」ではなくて「面倒だ」という感想をもたれるはずです。

なお，先へいくほど難しくなくなるという点で，プログラミングの性格は「学問」ではなくて，工作や機械操作と同じように，「技能」であることがわかります。

9 ケタあふれとケタ借り

16進数でFFに1を加えれば100になりますが，CPU内のアキュムレータは，AもBも2桁しかないので，ケタあふれとなり，答はただのゼロになってしまいます。

同様にゼロから1を引けば，ふつうはマイナス1に決まっていますが，アキュムレータではさっきの逆で，答はFFとなります。いわば架空の上位ケタからケタ借りを行って，100－1を計算しているわけですね。

実験で確かめてみましょう。**表10**のプログラムは，01から03を引いて，答を703F番に格納するものですが，RUNさせた後で703F番地を調べると，答はFEになっています。

図で表わすならば，アキュムレータAやBは，**図2**のような感じで計算をしていると思ってください。

10 機械語によるBEEP

ウォームアップの締めくくりとして，**表4**のBEEPのプログラムを，機械語で書いてみましょう。骨組みとしては，

(1)　BEEP ONですから，DB3F番地の

図2　アキュムレータの計算

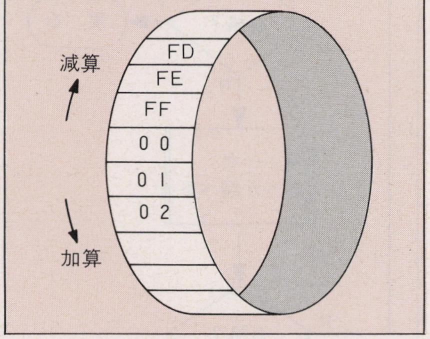


表10　「01—03」の計算

```
7000: 86 01       LET Acc.A=01
7002: 80 03       SUBTRACT 03 FROM A
7004: B7 70 3F    STORE A IN (703F)
7007: 39          RETURN
```

サブルーチンを使います。

(2)　タイマは7040番地から書くことにしましょう。

(3)　BEEP OFFします。

(4)　リターン

つまり機械語プログラムは**表11**のようになりますね。

そこでタイマですが，まず**表13**を見てください。アキュムレータBにFFを入れておいて，ゼロになるまで1を引いていきます。そうすると，FF回(10進数でいえば255回)だけ足踏みをして待っていることになりますね。

これだけでは機械語の実行が速すぎて，ぜんぜん音にならないので，さらに**表12**のとおりアキュムレータAを使って，さっきのタイマを20回(16進)だけ反復します。したがってアキュムレータBは，全体として10進数で255×32回だけ足踏みをすることになります。

主プログラムは**表8**のBASICと同じにしてください。RUNさせるとピッと音が出ます。7041番地の「20」を調節すれば，音の長さを変えることができます。

11 相対ジャンプ命令

いまのタイマで，「26　××」という相対ジャンプ命令を使っていますが，その××の書き方を，まだ説明していませんでした。

相対ジャンプ(68系では「ブランチ」と呼んでいます)においては，××の次の番地が基準になります。タイマでは「39」が書かれているところですね。この基準番地をゼロとして，そこからFF，FE，……と逆に数えていくと，××の値がわかります。さっきの**図2**と同じ数え方ですね。

検算をやってみましょうか。7082番地にブランチするとき，××はFDでしたね。この番地をFDとして，FE，FF，……と数えていくと，基準番地の7085番地のところで，ちょうどゼロになります。

番地の小さい方に戻るのでなくて，大きい方に進むときは，基準番地をゼロとして，01，02，……と数えるのですが，これはすぐあとでやってみましょう。

——以上でウォームアップを終わり，いよいよ実用プログ

表11　BEEPの機械語

```
7000: BD DB 3F    BEEP ON
7003: BD 70 40    TIMER
7006: BD DB 44    BEEP OFF
7009: 39          RETURN
```

表12　タイマ

```
7040: 86 20       LET Acc.A=&H20
7042: BD 70 80    SUB-TIMER
7045: 4A          LET Acc.A=Acc.A-1
7046: 26 FA       BRANCH IF NOT ZERO
7048: 39          RETURN
```

表13　タイマの下請けサブルーチン

```
7080: C6 FF       LET Acc.B=&HFF
7082: 5A          LET Acc.B=Acc.B-1
7083: 26 FD       BRANCH IF NOT ZERO
7085: 39          RETURN
```

ラムに挑戦です。

12 16進キー入力の概略設計

機械語プログラムやデータを入力するには，ふつうはモニタのMコマンドを使うのですが，0から9までの数字はともかく，AからFまでのキーは配置がバラバラで，あまり能率がよくありません。

さいわいキーボードの右側にテンキーがあるので，AからFまでについては，テンキーのまわりのキーを右回りに使うことにしましょう。すなわちABCDEFは，それぞれ＊／＋－＝，となります。

なおモニタのMコマンドでは，いちいちリターンキーを押しながら次へ進むのですが，どうもスピードが落ちてしまいます。機械語というのは，あとで必ず読み合わせをするのですから，書き込みのときはどんどん先へ進むようにしましょう。

13 フローチャートなど

人間が押すべきキーは，0から9までか，＊，／などです。正しいキーが押されたかどうかのマークは，アキュムレータBに担当させましょう。BがFFならば不合格，それ以外ならば合格とします。

それからBASICとの間でデータをやりとりするための窓口としては，707F番地を使うということにします。なお機械語の書き込み作業を完了したときは，ESCキーを押すという約束にしましょう。そうするとプログラムの構造は，図3のようになりますね。下請けサブルーチンに処理させることにします。

14 機械語の主プログラム

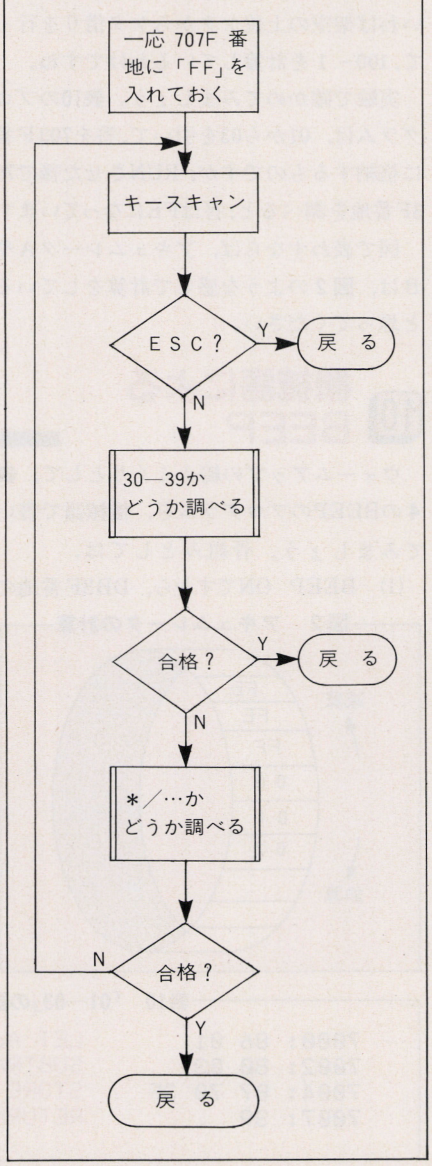
図3のフローチャートにしたがって，機械語プログラムを書いていきます。

(1) まずエラーマークをセットして，入力はいちおうエラーということにしておきましょう。アキュムレータBにFFを入れるのですね

(2) ついでに窓口の707F番地にも，FFを入れておきましょう

(3) 人間のキー入力を待ちます。キマリ文句のD072番地のサブルーチンを使うのですね

図3　16進キー入力の構造



(4) ESCが押されたら，リターンです。前回の実験によれば，ESCキーのキャラクタコードは1Bでしたから，アキュムレータAの値を調べて，1Bならばリターンのところにブランチします

(5) 押されたキーが0から9までかどうかを調べます

(6) アキュムレータBがFFでなければ，エラーではなく，正しいキーが押されています。リターンのところにブランチですね

(7) 合格でなければ，押されたキーが＊，／などかどうかを調べます

(8) またアキュムレータBを調べて，FFならばエラーですから(3)に戻るし，それ以外ならばリターンです

機械語は**表14**のとおりです。相対ブランチ命令がいくつか使われていますね。それぞれブランチ先が合っているかどうか，検算してみてください。

それから「比較命令」というのが3箇所ほど出てきています。具体例をお目にかけているので，使い方はだいたいおわかりと思いますが，基本的にはふつうの引き算と同じで，ただ命令実行後もアキュムレータの値は変わらないというものです。別の言い方をするならば，引き算を行ったあとで，アキュムレータの値を元に戻しておく命令といってもよいでしょう。したがって，この命令を使えば，アキュムレータの値を変えないで，引き算の結果がゼロかどうかを調べることができます。

15 0から9までのチェック

人間が0から9までのキーを押したかどうかを調べるプログラムを作りましょう。キャラクタコードで30から39までかどうかを見ればよいのですね。

(1) まず30と比較してみます。マイナスならばエラーですから，そのままリターンです

(2) 次に3Aと比較します。マイナスでなければ，アキュムレータAは3A以上ですから，やはりエラーです

表14　16進キー入力

```
7000: C6 FF        LET Acc.B=FF
7002: F7 70 7F     STORE B IN (707F)
7005: BD D0 72     INPUT 1 CHARACTER
7008: 81 1B        COMPARE A WITH 1B
700A: 27 0E        BRANCH IF ZERO
700C: BD 70 20     TEST IF 30-39
700F: C1 FF        COMPARE B WITH FF
7011: 26 07        BRANCH IF NOT ZERO
7013: BD 70 40     TEST IF 'A'-'F'
7016: C1 FF        COMPARE B WITH FF
7018: 27 EB        BRANCH IF ZERO
701A: 39           RETURN
```

表15　0から9までのチェック

```
7020: 81 30        COMPARE A WITH 30
7022: 25 0A        BRANCH IF CARRY
7024: 81 3A        COMPARE A WITH 3A
7026: 24 06        BRANCH IF NO CARRY
7028: 80 30        SUBTRACT 30 FROM A
702A: B7 70 7F     STORE A IN (707F)
702D: 5C           LET B=B+1
702E: 39           RETURN
```

表16　*,／などのチェック

```
7040: C6 0A        LET B=0A
7042: 81 2A        COMPARE A WITH 2A
7044: 27 1B        BRANCH IF ZERO
7046: 5C           LET B=B+1
7047: 81 2F        COMPARE A WITH 2F
7049: 27 16        BRANCH IF ZERO
704B: 5C           LET B=B+1
704C: 81 2B        COMPARE A WITH 2B
704E: 27 11        BRANCH IF ZERO
7050: 5C           LET B=B+1
7051: 81 2D        COMPARE A WITH 2D
7053: 27 0C        BRANCH IF ZERO
7055: 5C           LET B=B+1
7056: 81 3D        COMPARE A WITH 3D
7058: 27 07        BRANCH IF ZERO
705A: 5C           LET B=B+1
705B: 81 2C        COMPARE A WITH 2C
705D: 27 02        BRANCH IF ZERO
705F: C6 FF        LET B=FF
7061: F7 70 7F     STORE B IN (707F)
7064: 39           RETURN
```

表17　16進キー入力(総合)

```
200 REM *** HEX KEY-IN ***
210 CLEAR 300,&H6FFF
220 DEFINT A,D,F,I,V
230 DEF USR0=&H7000
240 CLS
250 INPUT "START ADDRESS";A$
260 A=VAL("&H"+A$)
270 F=0
400 PRINT HEX$(A);": ";
410 FOR I=0 TO 7
420 GOSUB 2000:'INPUT 1 BYTE
430 IF F>0 THEN END
440 POKE A,D
450 A=A+1
460 IF I=3 THEN PRINT "  ";
470 NEXT I
480 PRINT
490 GOTO 400
2000 REM *** INPUT 1 BYTE ***
2010 V=USR0(V)
2020 V=PEEK(&H707F)
2030 IF V=255 THEN F=1:RETURN
2040 PRINT HEX$(V);
2050 D=V*16
2060 V=USR0(V)
2070 V=PEEK(&H707F)
2080 IF V=255 THEN F=1:RETURN
2090 PRINT HEX$(V);" ";
2100 D=D+V
2110 RETURN
```

(3)　両方の試験に合格したときは，Aから30を引いて，707F番地に入れてやります

(4)　アキュムレータBをFF以外の値にします。1を加えてやりましょう。Bはちょうどゼロになるわけですね

プログラムは**表15**です。

この中で「キャリー」というのは，聞いたことのない方も多いと思いますが，これは引き算（比較命令でも同じ）において，答がマイナス（ケタ借り）になったことを意味します。

16　*,/などのチェック

最後のプログラムは**表16**で，ちょっと長いですが，新しい命令は1つも出てきません。

(1)　アキュムレータBを，いちおう0Aにしておきます

(2)　押されたキーが*（キャラクタコードで2A）かどうか調べます

(3)　そうであれば最終処理へ

(4)　Bに1を加えます。答は0Bになりますね

(5)　押されたキーが／かどうか調べます

(6)　以下同様に，"+"，"−"，"="，"，"かどうか調べます

(7)　どれかに合致しているとき，アキュムレータBはそれに対応する値をとっていますね。*ならば0Aだし，／ならば0Bです。どれにも合致していなければ不合格ということになりますから，Bの値を再びFFに戻します

(8)　Bを707F番地にストアして，リターンします

17　BASICプログラム

BASICは説明するまでもないでしょうが，**表17**のとおりで，

(1)　第1のキー入力の数値を取ってくる

(2)　それを16倍して，第2のキー入力の数値を加える

(3)　合成された数値をメモリにPOKEするという作業の繰り返しです。

なお707F番地が255（16進数ではFF）のときは，ESCキーが押されたのですから，作業終了です。

また，**表17**をRUNする前に必ず，**表14，15，16**の機械語はメモリに入れておいてね。

＊＊　　＊＊　　＊＊

正直なところ，16進キー入力ぐらいならば，BASICだけでも書けないわけではありませんが，読者は機械語を見かけよりは簡単だなと感じていただけたでしょうか。

後半戦では機械語の命令を少しふやして，「チェックサム」と「メモリ内容の移動」のプログラムを書いてみましょう。

SOFT REVIEW

①ジャンル
②提供媒体（T：カセット，TB：カセットブックタイプ，5D：5インチディスケット，8D：8インチディスケット）
③価格
④適用機種
⑤使用言語
⑥必要なオプション
⑦発売元

SOFT REVIEW

デゼニランド

①アドベンチャーゲーム
②T,5D(マニュアル付)
③T：4,800円　5D：6,800円
④FM-7
⑤マシン語
⑥なし
⑦ハドソン

東京ディズニーランドをパロったアドベンチャーゲーム。1歩デゼニランドの中に入ると，そこには行ったことのある人にしかわからない東京ディズニーランドの風景が広がるのです。ストーリーは，千葉県と埼玉県の血で血を洗う抗争の末，建設されたデゼニランド，その中に隠された「三月磨臼」という純金製の臼を捜す，といったもの。アドベンチャーに慣れている人でも，解決までに1月ぐらいかかるのでは？　暗いゲームの多い中で，明るく健康的なゲームです。

特筆すべきこととして，ペイントが早い。ペイントの遅いアドベンチャーゲームほどつまらないものはありませんが，この「デゼニランド」はペイントの遅さにいらいらしなくてもよいのです。これはテープ版でもディスク版でも同じ。それに114画面という画面構成がすごい（たいていのゲームは50画面ぐらいです)。中間色ペイント，なおかつ高速，というものなのです。

ゲーム自体は少々難しいかも。場面ごとに必要なものを見極めないと，デゼニランドの中でうろうろするだけになってしまいます。ゲームはSAVEできるので，気長にどうぞ。

EARTHBOUND

①アドベンチャーゲーム
②TB（マニュアル付）
③4,500円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦クリスタルソフト

ある朝目を覚ますと巨大な毒虫になっていた，んじゃなくて，記憶喪失になっていた。たったひとつ思い出せるのは“レディ・ジューン”という名前だけ。あなたはラ・テールの村に行き，レディ・ジューンを捜し出さなければならない。

タイトルグラフィックがいい。とても不思議な雰囲気をもったゲームで，思わずひきずりこまれてしまう。大人が遊ぶのにもいい。ゲーム自体はやや難しい。

ミステリーハウスI

①アドベンチャーゲーム
②TB（マニュアル付）
③3,800円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦マイクロキャビン

たくさんあるアドベンチャーゲームでも，“元祖アドベンチャーゲームを”という人には，この「ミステリーハウス」をおすすめ。

この家には宝物が隠されている。あなたはその宝物を捜しだし，持ち出さなくてはならない。コンピュータは“COMMAND？”という具合で聞いてくるので，英語の“動詞＋名詞”で答える。そして，いろいろある部屋の中を回って，宝物を探してほしい。

簡単そうですが，なかなか難しいですよ。推理小説を読むような気分になれると思います。

画面はきれいでシンプル，ゲーム進行もスムーズ。頑張ってください。

不思議な旅

①アドベンチャーゲーム
②TB（マニュアル付）
③3,600円
④FM-7,8(共用)
⑤マシン語＋BASIC
⑥なし
⑦エニックス

SF的アドベンチャーゲームです。

終末戦争のため地球人類はほぼ死滅し，君は幻想の世界へと投げとばされる，というのがストーリー。気がつくと君の回りには誰もいない。そして君は，地球へ戻るときに，とても不思議な旅に遭遇する。

このゲームは常識的な考えだけでは，なかなか解けません。次々とむかえる意外な新局面をどう解決するか，大胆な発想と明晰な頭脳が要求されます。あなた自身がこの世界にのめりこむことが大切ですよ。

FMならではの色を楽しめるグラフィックです。現実とは違う幻想的な気分にひたるのを楽しんでみては？

4次元の家

①アドベンチャーゲーム
②T（マニュアル付）
③3,800円
④FM-7
⑤BASIC
⑥なし
⑦プロシューマー

ここは村はずれ。そこには不思議な建物が。それは4階建ての家で，2階には四方に立方体の部屋が張り出している。各部屋には窓が1つとドアが1つ。そして今，私の目の前で家は異様に変形し，2階以上が見えなくなってしまった。私は調査のため単身，この家に入ってみたが……。

Lovely飛鳥

①アドベンチャーゲーム
②TB（マニュアル付）
③3,200円
④FM-7
⑤BASIC
⑥なし
⑦エニックス

飛鳥ちゃんに会いたくて町を右往左往。多くの難関を突破しつつ，やっと彼女に会えた。そのごほうびは？

ゲームを進めていると，クイズを出題されます。正解じゃないとENDになってしまうから，気をつけて。

ロリータ・シンドローム

①アドベンチャーゲーム
②TB（マニュアル付）
③3,600円
④FM-7
⑤マシン語＋BASIC
⑥なし
⑦エニックス

ロリコンでグロ好きの人にうってつけ。絵は可愛いけれど，内容は残酷です。

PIPE MAN

①アクションゲーム
②TB
③3,400円
④FM-7,8(共用)
⑤BASIC＋マシン語
⑥（ジョイスティックでプレイする場合）
ジョイスティック（BYTE SHOP製およびその相当品）
⑦システム・イン・猫

あなたは水道工事屋「PIPE MAN」。イタズラモンスター「ヘラヘラ」がかじって水もれしているところを直さなくてはならない。しかし，ヘラヘラが邪魔をする。ヘラヘラの攻撃をかわしながら，時間内に水をいっぱいにするのです。画面は全部で3画面あり，それぞれの画面でヘラヘラは4,6,8匹と増えていきます。

PIPE MANの動き，シャワーを浴びているペンギンなど，キャラクタが可愛く，楽しいゲームを求めている人には，おすすめソフト。ただし，キャラクタが可愛いからといって，見とれないように。あっという間にヘラヘラに食べられてしまいます。簡単そうに見えるけれど，これがなかなか……なのです。熱中できるゲームといえます。また，ジョイスティックでのプレイも可能です。

みゆき THE 勝負師

①ゲーム
②T（マニュアル付）
③3,200円
④FM-7
⑤マシン語＋BASIC
⑥なし
⑦パックス・エレクトロニカ・ジャパン

このゲームソフトは，真剣な勝負をしながらも，少々ユーモアがほしい，という人にピッタリのものです。

ゲームは，ほとんどポーカーに似ていて，“みゆき”というかわいい女の子とあなたが勝負するのです。積んであるトランプの山から，みゆきとあなたが交互に1枚ずつとっていって，5枚同じマークを揃えます。そしてお互い揃ったカードで，点数を争うために山からとっていきます。

それから，ポーカーの要領で，お互いのかけひきでかけ金をつりあげていきます。あなたが，give upするとあなたの負け。点数が少なくても，あなたの負け。

もし，みゆきが持ち金0(ゼロ)になると，みゆきの服と交換に2万円貸します。つまり，あなたが勝ち続けると，みゆきは洋服を……。あとは，お楽しみです。

ゲーム進行もスムーズで，色彩豊かなきれいなグラフィックです。かわいい音楽も流れていいですよ。

BLITZ KRIEG
（ブリッツ・クリーク）

①シミュレーションウォーゲーム
②T
③4,500円
④FM-7
⑤BASIC
⑥なし
⑦マイコンセンターウエノ

本格派戦争ゲームが，登場しました。とても頭脳を使う，じっくり考えるゲームで，どんな年齢の人でも，楽しめるものです。

自分の隊には，いろいろな役目をもった戦艦，戦闘機，歩兵隊……などがあり，うまく操って相手を攻撃するのです。しかし，戦闘機などにも，燃料の制限がありますし，相手もなかなか見つからなかったりして，あなたの推察力や根気がためされますよ。じっくり，楽しみたい人は，地図などを書いてから，とり組むと，より一層よいと思います。画面が少々寂しい感じがしますが，それは，こっているゲームの展開が埋めてくれます。

さあ，あなたも戦場の指揮者です。あなたの作戦で，敵をたおしてください。

陸軍中野学校

①アクション＋思考型ゲーム
②T
③3,800円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦東映ビデオ

あなたはスパイとして，猛訓練に耐え，奇酷な任務を遂行しなければならない。はたして優秀なスパイになれるか。

筒井康隆の「ケンタウルスの殺人」

①ゲーム
②T
③3,600円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦東映ビデオ

筒井康隆原作「ケンタウルスの殺人」のパソコンゲーム化。

悪女かまきり

①ゲーム
②T（マニュアル付）
③3,900円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦東映ビデオ

あなたは悪女かまきり。パトロンの堂島を殺して，うまく保険金を手に入れられるか。ピンクのカセットテープがいい。

スーパーコンボイ

①ゲーム
②TB（マニュアル付）
③3,900円
④FM-7
⑤BASIC
⑥なし
⑦東映ビデオ

あなたはハンドルひとつに命をかける一番星の桃次郎です。あこがれのマドンナにプロポーズするために，資本金を50万円から1,000万円にしてください。途中，さまざまなアクシデントが起ります。シミュレーション的なゲーム。

噴火サバイバル

①アドベンチャーゲーム
②T（マニュアル付）
③3,800円
④FM-7，8（共用）
⑤BASIC＋マシン語
⑥なし
⑦CSKソフトウェアプロダクツ

突然，火山が噴火した。なんとかふもとの

町までたどり着かなければならない。
ゲームといえども，近い将来あり得ることです。真剣にプレイしましょう。

智将ロンメル

①シミュレーションゲーム
②TB（マニュアル付）
③4,600円
④FM-7
⑤BASIC
⑥なし
⑦CSKソフトウェアプロダクツ

ロンメル将軍率いるドイツ・アフリカ軍国の戦いをテーマにしたもの。プレイヤーは枢軸軍（ドイツ・イタリア）を指揮し，英軍と戦います。冷静な頭脳でどうぞ。

VCヘリボーン作戦

①リアルタイムシミュレーション
②T（マニュアル付）
③4,800円
④FM-7
⑤マシン語＋BASIC
⑥なし
⑦東映ビデオ

ベトコンの活動を封止せよ。ベトコンを探し出し，殲滅せよ。これが君に与えられた任務だ。恐怖と混乱が支配する戦場で君は勝つことができるだろうか。

タートル

①思考型ゲーム
②T（マニュアル付）
③2,800円
④FM-7,8（共用）
⑤BASIC
⑥なし
⑦ストラットフォード，コンピューターセンター

六角形の盤上で争う，新しいオセロゲーム。61コマをどう取るか？　コンピュータの判断は正確ですばやい。

ヨットレース

①シミュレーションゲーム
②TB（マニュアル付）
③3,800円
④FM-7,8（共用）
⑤BASIC＋マシン語
⑥なし
⑦ライブハウスアロー

静岡県焼津港〜千葉県館山港のヨットレースを，グラフィックでシミュレーションしたもの。ヨットは実際と同様にメインセール，ジブセールを風向き，風速に応じ操作することにより前進する。ヨットの操作方法は最初は少々難しいが，マニュアルをよく読んでチャレンジ。あなたのヨットは赤。マイコン1号，2号と3艇のヨットでスピードを競う。冷静な判断力が必要。どちらかといえば，ヨットを知っている人に。

STAR STREAM

①ゲーム
②T
③3,200円
④FM-7
⑤マシン語
⑥なし
⑦ハドソン

宇宙嵐(スターストリーム)にまき込まれた地球防衛軍パトロールは，好戦的なバルタン星人の領域深く舞い込んでしまった。バルタン星人のスクランブルを受けながらも，パトロールはセンサを頼りに敵のマザーシップを発見し，撃墜しなければならない。

反射神経型ゲームの緊迫感とキャラクタの可愛さがミックスされたゲーム，画面がスクロールするので迷子にならないように！

BEAN'S JACK

①ゲーム
②T（マニュアル付）
③3,200円
④FM-7
⑤マシン語
⑥なし
⑦ハドソン

「ジャックと豆の木」風のストーリー。魔法の豆の木の種を手に入れたジャックは，カラスの攻撃をかわしながら豆の木を育てます。次に彼は，大きくなった豆の木に登って雲の上にいる巨人の宝物をとりにいきますが，無事に地上へ戻ってこれるでしょうか。手先の器用さが要求されるゲーム。

キャプテンスミス

①ゲーム
②TB（マニュアル付）
③2,800円
④FM-7
⑤マシン語
⑥なし
⑦MIA

宇宙歴2001年，幾多の戦闘を繰り返し，数々の戦果をあげてきたキャプテン・スミス。今日もスミスに，緊急指令が送られてきた。

3次元的なセンスが必要なゲームです。敵をうまくキャッチしないと，宇宙の中に取り残されたような気がしてきます。

コスモクラッシュ

①反射神経型ゲーム
②TB（マニュアル付）
③3,200円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦エニックス

これは反射神経型ゲームで，TVゲームなどによく見られる感じのものです。

あなたは宇宙を飛行しています。隕石や敵機をよけて上手に飛行しなくてはいけません。おまけに敵機からはミサイル攻撃をうけるので，応戦しながら飛行します。全部で10画面構成となっています。あなたはどのくらいクリアできるでしょうか。ハイスコアめざして，点数を競ってください。

画面は宇宙の神秘さを表現した，すてきなゲームです。

GAME PACK

①ゲーム
②5D
③5,800円
④FM-11
⑤BASIC
⑥なし
⑦マイクロキャビン

「MICRO MOUSE」,「EXCITING GOLF」,「MIDNIGHT COMMANDER」の3本のゲームのセット。

MICRO MOUSEは，三次元迷路を走り抜けてゴールするゲーム。EXCITING GOLFは，本格的なゴルフシミュレーション。MIDNIGHT COMMANDERは，シミュレーションウォーゲーム。

ABYSS

①アドベンチャーゲーム
②5D
③9,800円
④FM-7,8
⑤BASIC＋マシン語
⑥なし
⑦エム・エー・シー

深宇宙調査員である私は，とある星の植民星化調査中，この星に宇宙犯罪シンジケートの基地が存在することを発見した。そ

して，その星に関する記憶を一切消去されてしまう。私の目的は，いち早くこの星の座標をさぐりあてて，九惑星警察に通報することである。

スターライトアドベンチャー パート２

①アドベンチャーゲーム
②ＴＢ
③3,800円
④FM-7,8(共用)
⑤マシン語
⑥なし
⑦キャリーラボ

主人公は彼の宇宙船で８つの惑星を駆け巡り，プリンセスを救い出さねばなりません。ゲームは主人公の行う報告や疑問に対して，あなたが指示を与える，という形で進行します。しかし彼は，あなたの指示をあおがずに勝手に行動することもあります。

フライトシミュレーター スペースビー

①リアルタイムゲーム
②ＴＢ
③4,000円
④FM-7
⑤マシン語
⑥なし
⑦キャリーラボ

ジェルダ

①ゲーム
②ＴＢ
③3,800円
④FM-7
⑤マシン語
⑥グラフィックボード
⑦キャリーラボ

ウルトラセブンブリッジ

①思考型ゲーム
②ＴＢ
③3,500円
④FM-7,8
⑤マシン語
⑥なし
⑦キャリーラボ

コンピュータを相手の本格的カードゲーム。グラフィック機能を生かした美しい画面です。パソコンが３人分プレイしますので，あなたとパソコンの４人でプレイします。

倉庫番２

①思考型ゲーム
②Ｔ，５Ｄ
③Ｔ：4,300円　５Ｄ：4,900円
④FM-7
⑤BASIC＋マシン語
⑥なし
⑦シンキング・ラビット

前号で紹介した「倉庫番」のパート２。遊び方は前と同じだけど，今回は倉庫が50面。かなりレベルアップしている。また，“倉庫番エディタ”で，自分で問題を作成できる！　かなりの充実度，といえるだろう。

タロット占い

①ホビー
②５Ｄ
③5,000円
④FM-7,8
⑤BASIC
⑥(必要な場合)プリンタ
⑦パソコンハウストム

この「タロット占い」は，大アルカナ22枚を使って占います。現代社会の中に真の存在を見失いつつある私たちの，日常の生きざまをその不思議な絵解きの世界で占ってみましょう。

王　将

①ホビー
②Ｔ,5Ｄ
③Ｔ：4,000円　５Ｄ：6,000円
④FM-7
⑤マシン語
⑥なし
⑦キャリーラボ

マイコンでついに実現。詰将棋ではありません，本当に闘うのです。３～５級程度の実力あり。

グラフィック・キャンバス

①グラフィックツール
②Ｔ，５Ｄ（マニュアル付）
③Ｔ：3,900円　５Ｄ：9,800円
④FM-7,8,11(共用)
⑤BASIC＋マシン語
⑥（必要な場合）デジタイザ（関東電子K510），プリンタ
⑦アイ企画

操作はメニュー方式によるコマンド選択で，非常に簡単に使用できる。入力方法はキー入力（カーソル移動）かデジタイザ入力かを選択でき，細かな表現を可能にした。簡単に中間色（３色までの組み合せ）をペイントできる。中間色は全60色（40色は登録済み。20色はユーザー登録可能）。作成したグラフィックはマシン語で実行するので，高速処理が可能。

たのしいオリガミPART１～３

①学習
②Ｔ
③各3,000円
④FM-7
⑤BASIC
⑥なし
⑦T.S.Sスリー・スターズ・ソフト

「幼児向学習ソフトシリーズ」の第１弾。メニュー画面より折りたいものを選択し，グラフィックと説明文により折っていく。

PART1では①ツル，②セミ，③ホカケブネ，④フウセン，⑤カメ，⑥カブト，⑦カニの７種類の折り紙と，⑧たのしいオリガミのうたを収録。PART2では①イカ，②クジャク，③イエ，④ネッタイギョなど，PART3では①ハト，②コトリ，③イス，④ウミガメなどを収録。

たのしい英絵単語

①学習
②Ｔ
③3,000円
④FM-7
⑤BASIC
⑥なし
⑦T.S.Sスリー・スターズ・ソフト

「幼児向学習ソフトシリーズ」の第２弾。水色ＵＦＯを操作して答えを選び，赤色ＵＦＯの判定を待つ。①和英のテスト，②英和のテスト，③勉強（アルファベット順に楽しい絵がついた英単語が表示される），④たのしい英絵単語のうた。

英絵単語データPART１(A～D)～PART４(P～Z)

①学習
②Ｔ
③各2,000円
④FM-7
⑤BASIC
⑥なし
⑦T.S.Sスリー・スターズ・ソフト

「たのしい英絵単語」の拡張データソフト。PART1はA～Dで始まる英単語を70ワード収録。PART2はE～Hで始まる英単語70ワード。PART3はＩ～Ｏで始まる英単

語70ワード。PART4はP～Zで始まる英単語70ワードを収録。

まんてんくん

①学習
②TB（マニュアル付・テープ3巻）
③9,800円
④FM-7
⑤BASIC
⑥漢字ROM
⑦ランドコンピュータ

幼児用学習ソフト。数えることの学習，10の合成・分解，120までの数の3段階を選択できる。日々の学習成果は，付属の記録用紙に記入できるようになっている。グラフィックが多く使われている。

中学1年数学ホイホイ

①学習
②TB（マニュアル付）
③3,500円
④FM-7,8(共用)
⑤BASIC
⑥なし
⑦ベーシックシステム

1本のテープに3本のソフトが入っており，自分のレベルに応じて選択できる。

上巻は整数，公約数，公倍数，加法・減法など，中巻は正の数・負の数の特訓，下巻は方程式，比例・反比例などから成っている。練習問題は解法と答を出し，♥で隠された部分を当てる，という方法をとっている。画面がシンプルで見やすい。

中学2年数学ホイホイ

①学習
②TB（マニュアル付）
③3,500円
④FM-7,8(共用)
⑤BASIC
⑥なし
⑦ベーシックシステム

1本のテープに2本のソフトがおさめられている。上巻は1次不等式，連立不等式など，下巻は1次関数，連立方程式など。解答の仕方は中学1年生用と同じ。

中学3年数学ホイホイ

①学習
②TB（マニュアル付）
③3,500円
④FM-7,8(共用)
⑤BASIC
⑥なし
⑦ベーシックシステム

3本のソフトがおさめられている。上巻は因数分解，平方根，平方根を含む式の計算など，中巻は三平方の定理，2次方程式など，下巻は確率，統計などから構成されている。解答の仕方は，中学1年生用と同じ。

FMワードV3.0

①ワープロ
②5D（マニュアル付）
③19,000円
④FM-7，8，11
⑤BASIC＋マシン語
⑥プリンタ，漢字ROM
⑦綜合資料研究社

テキストは50桁×45行で構成。文章入力時に編集ができる。漢字変換はカナ1桁～6桁の入力で，カーソル位置に表示する方法で行う。辞書ディスクは28,700語登載，オプションによりユーザー登録13,300語まで。文書ディスクはフルにテキストを使用した状態で50ファイル登録可能。短文はオプションで登録，1ディスケットあたり1短文120字で1,200件まで可能。

印字文字種は，半角・全角・倍角が可能で，画面にもそれぞれ表示する。印刷には初期設定を行うが，印刷時の変更も可能。そのほか関連した文書の連続印刷（ページ付），または単独印刷ができる。

フロッピはシングルでも大丈夫。

PN101

①ワープロ
②5D（マニュアル付）
③18,000円
④FM-7
⑤アセンブラ＋BASIC
⑥漢字ROM，プリンタ
⑦エス・ピー・シー

連続文書一括入力により，一度に1ページの文書をブラインドタッチで入力できる。専用機に匹敵するような編集機能を備え，あらゆる用途に対応できる。棒線，罫線，アンダーラインなどを自由に引くことができ，表や棒グラフなどが簡単に作図できる。コマンドはファンクションキーに設定し，画面下に表示されているので画面を見ながらワンタッチで操作できる。

WORD KING

①ワープロ
②5D（マニュアル付）
③18,000円
④FM-7　⑤マシン語＋BASIC
⑥漢字ROM，フロッピ2ドライブ，プリンタ（RP-80，FP-80，他ただしMB27404を除く）
⑦CSKソフトウェアプロダクツ

入力方法は，カナ漢字変換，ローマ字変換のいずれかを選択できるため，カナタイプに不慣れな人でも入力はスムーズ。倍角・半角文字はもちろん，罫線・アンダーラインなども引くことができる。辞書には約2万語が登録済みだが，ユーザーによる127文字以内の略語・短文登録も可能。暗証番号による文書登録を行うので，文書の機密保持に役立つ。入力した文字はブルーで表示され，CRTを長時間見ていても，眼が疲れない。

COMAS給与

①給与計算
②5D，8D（マニュアル付）
③5D：99,800円　8D：135,000円
④FM-7，8(共用)，11
⑤アセンブラ
⑥漢字ROM，フロッピ2ドライブ
⑦コマス

漢字データの入力が，カーソル位置におけるカナまたはローマ字変換で行え，計算処理が高速（1人当りの計算所要時間は約10秒，100名で約2分）である。また，ファイルディスケットを替えることにより，各種の管理資料や計算処理ができる（データベースソフト「COMAS-K2」を内蔵）。

500名までの企業で利用できる。なお，オプションで年調と賞与計算があり，価格はセットで20,000円。

給与計算

①給与計算
②T（マニュアル付）
③4,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

月給・日給・パートの給与計算や，3回の賞与支給，1年間の労働実績の管理ができるほか，個人の給与明細書，給与台帳の印刷ができる。人数は20人を単位として何人でも計算可能。給与明細書には個人別金種表も合せて印刷される。

スーパー漢字住所録

①顧客管理
②T，5D　（マニュアル付）

③12,500円
④FM-7
⑤BASIC＋マシン語
⑥漢字ROM，プリンタ（RP-80，FP-80，MB27406，GP-250F，他）
⑦日本ファルコム

カナ２文字によるカナ漢字変換方式。初めての人にも簡単に使える。検索，ソート，データ入出力などの高速化を計るため，マシン語を使用している。検索シートなどが180件までオンメモリで処理できるので，大量のデータを高速に取り扱える。住所の一覧表，電話番号表，あて名書きが，グラフィックプリンタでも漢字プリンタと同様にきれいに印字できる。

情報管理

①データベース
②T（マニュアル付）
③4,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

項目数は10項目。項目名はユーザーにより自由に設定でき，複数の検索条件によって該当する情報を画面表示，またはプリントアウトできる。市販のハガキメール用シールのあて名書き機能がついているので，顧客管理，住所録，図書検索および各種の情報管理に利用できる。１画面を１ページとして100ページまでの処理が可能。

見積書作成

①見積
②T（マニュアル付）
③4,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

「実戦！　在庫管理」「実戦！　販売管理」「実戦！　仕入管理」のいずれかのプログラムで作成した商品元帳のデータと共用できるほか，本プログラムで作成した商品元帳のデータで見積書を作成する。見積書に記入できる商品数は20までで，作成した見積書はプリントアウトやカセットテープに記録することができる。

実戦！　売掛金管理

①売掛
②T（マニュアル付）
③4,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

「実戦！　販売管理」や本プログラムで作成した得意先元帳によって，売掛金の管理を行う。得意先元帳や入金日計表，売上・売掛日計表，売掛残高一覧表のプリントアウトが可能なので，資金繰りに活用できる。

新住所録

①ビジネス
②５D（マニュアル付）
③7,500円
④FM-7，8，11
⑤BASIC＋マシン語
⑥プリンタ
⑦綜合資料研究社

住所録データは１ディスケットあたり，512件。１データあたり２本のインデックスファイルを設定，検索はそのインデックス，または１データ全体に対しても実行できる。テキストは６×33文字で構成。これにより小規模のデータベースに利用できる。住所録データは連続的に画面表示されるが，キーコードにより表示の禁止もできる。

辞書ディスクは28,700語登載，オプションによるユーザー登録は13,300語まで。印字文字種は半角・全角・倍角が可能で，画面にも表示する。ハガキを縦２列に連続印刷，タックシール印刷も可能。

実戦！　株価分析

①ビジネス
②５D，８D（マニュアル付）
③５D：15,000円　８D：20,000円
④FM-7，8，11
⑤BASIC
⑥（コピーをとる場合）プリンタ
⑦近畿コンピュータサービス

400日分の株価・出来高を管理して，星足・移動平均線・出来高・逆ウオッチ曲線の各グラフを作成。ミクロからマクロまで株価を分析できる投資家向けソフト。５インチディスケット１枚で，250銘柄の管理・分析ができる。

転換社債分析SOFT

①ビジネス
②５D（マニュアル付）
③9,800円
④FM-7，11
⑤BASIC
⑥プリンタ
⑦ティー・アイ・ピー

転換社債の「買い」の診断をコンピュータで行う。さまざまな判断するための数値を表示し，グラフ化するので，自分で判断するときの手助けともなる。

ヘルパー

①ビジネス
②TB（マニュアル付）
③8,000円
④FM-7，8
⑤BASIC
⑥プリンタ
⑦シナプス１

個人商店向き（製造業を除く）の青色申告用ソフト。簡易型現金出納帳（複式準拠），簡易型決算整理記入の２本のプログラムより構成。経費科目分類表，経営診断チェックなどの参考資料を活用すれば，不慣れな人でも最小のシステムで経営管理ができる。

写植レイアウトプログラム

①ビジネス
②５D（マニュアル付）
③268,800円
④FM-7，8，11（共用）
⑤BASIC
⑥漢字ROM，（コピーをとる場合）プリンタ
⑦アンプルメディア販売

誌面レイアウト（行数，ミリ），級数歯送り表，原稿用紙印刷，ミニレイアウト印刷の５つのプログラムで構成され，従来のような煩雑な計算なしに割付指定が行える。各プログラムのデータは互換性があり，レイアウトプログラムでレイアウトしたものを，級数歯送り表プログラムで割付し，ミニレイアウト印刷プログラムでレイアウト用紙の印刷，といった一貫性のある作業が可能。入力したデータはすべてカラー表示されるので，誌面イメージを確実に捉えることができる。また，縦横とも10mまで計算できるので，どんなサイズでもレイアウトできる。

FM CALC

①簡易言語
②T，５D（マニュアル付）
③T：15,000円　５D：48,000円
④FM-7，8
⑤マシン語
⑥なし
⑦ソフトマート

FM CALCとは作表用の簡易言語で，コンピュータ用言語を知らない人でも簡単

に操作できる。作表機能のほか，集計，ソートなどの機能も備わっており，ビジネスの場に最適。

FM-積算PART I 軀体編

①科学技術
②5D（マニュアル付）
③300,000円
④FM-11
⑤BASIC
⑥漢字ROM，増設フロッピ，プリンタ
⑦ケンブリアランゲージサービス

軀体部建築積算。短時間で正確な積算値が得られる。データ入力は対話方式。FM-7，8用のものに比べ，登録件数が増え，処理スピードが大幅アップ。

FM-積算PART II 鉄筋編

①科学技術
②5D（マニュアル付）
③300,000円
④FM-11
⑤BASIC
⑥漢字ROM，増設フロッピ，プリンタ
⑦ケンブリアランゲージサービス

実長計算算，定尺換算により鉄筋注文表，加工表を作成。総トン数，効率的加工の際のロス量まで計算。その他，必要なデータを詳細に渡り算出。軀体と鉄筋が1日で積算可能。

計算式プログラム

①科学技術
②5D（マニュアル付）
③50,000円
④FM-7，8（共用）
⑤BASIC
⑥漢字ROM,フロッピ2ドライブ,プリンタ
⑦ケンブリアランゲージサービス

「FM-積算PART I 軀体編」で算出された各箇所の値の計算過程を表示する。データの確認，検算には欠かせない。

グラフパック20

①グラフ作成
②T,5D（マニュアル付）
③T：4,900円　5D：5,900円
④FM-7,8,11（共用。ただし11はプログラムを修正）
⑤BASIC
⑥プリンタ(MB27401および相当品)
⑦コンピュータのアルファ

経営管理などのビジネスの現場に役立つ20のグラフと，10の分析表を，データ入力だけで自動的に作成する。データを一度，汎用入出力プログラムによるデータ表に入れておけば，何回でも必要な部分を取り出してグラフ作成できる。グラフの縮小，スケーリングは自動的に行われ美しい。ハードコピーをとれば，そのまま実務に使用できる。作成できるグラフは折れ線グラフ，帯グラフなどのほか，レーダーチャート，商品特性分析グラフ，損益分析点グラフ，販売戦略グラフ，X-Yプロットグラフなど。

実戦！　QC管理(6)～(9)

①グラフ作成
②T（マニュアル付）
③各5,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

X管理図および$\overline{X}$－R管理図などを作成。データシートをプリントアウトできる。カセットテープへのデータの保存もできる。

実戦！　QC管理(A),(B)

①グラフ作成
②5D，8D（マニュアル付）
③5D：各20,000円　8D：各22,000円
④FM-7，8，11
⑤BASIC
⑥プリンタ
⑦近畿コンピュータサービス

特性要因図，パレート図，散布図，ヒストグラム，棒グラフ，折れ線グラフ，円グラフ，レーダーチャート，帯グラフ，ABC分析の各種グラフなどを作成。グラフ数は6～12枚を単位として，ディスケット記録容量いっぱいまで可能。データシートへのプリントアウトもできる。

FLEX SUPPORT PAC-7，8，11

①FLEXユーティリティ
②5D（マニュアル付）
③FM-7，8用：30,000円
　FM-11用：25,500円
④FM-7，8，11
⑤マシン語
⑥なし
⑦ソフトマート

FM用のFLEXをサポートする。モニタコマンド（メモリダンプ，メモリ表示および変更，ユーザープログラムの実行など），シングルディスクコピーコマンド，セクタイメージコピーコマンド（ボリュームコピー），F-BASICとのディスクフォーマット変換コマンド，BIOSを含む。

αカード＋CP/M68K

①OS
②αカード＋5D（マニュアル付）
③198,000円
④FM-11
⑤C
⑥なし
⑦ソフトマート

αカードとCP/M68Kのセット。

ROM WRITE Utility

①FLEXユーティリティ
②5D
③15,000円
④FM-7，8，11
⑤アセンブラ
⑥なし
⑦ソフトマート

FLEX下でオブジェクトをROM化するためのユーティリティ。TR4941；ベッカー用。

ラムディスク(CP/M-86)

①CP/M-86ユーティリティ
②5D
③16,000円
④FM-11
⑤マシン語
⑥拡張RAMボード
⑦カーネル

CP/M-86において，拡張RAMを半導体ディスクとして使用できるようにする。拡張RAMを128Kバイト単位で，最大768Kバイトまで使用することで超高速アクセスドライブが実現。

68000 DisAssembler on 6809

①FLEXユーティリティ
②5D（マニュアル付）
③50,000円
④FM-7，8，11
⑤マシン語
⑥なし
⑦ソフトマート

ショートプログラム集

生き残りゲーム——塩見　芳章（東京都杉並区）　for FM-7,8,11

よくあるゲームです。2人で遊びます。障害物と，相手および自分の通ったあとにぶつかると死にます。生き残ったほうの勝ち，というゲームです。

RUNすると，まず2人の名前を聞いてくるので，適当に入力してください。それが終わると，しばらくの間障害物を作ります。そして，双方のドットがスタートするので，自分のほうを次のキーでコントロールしてください。

右の人……[2]，[4]，[6]，[8]

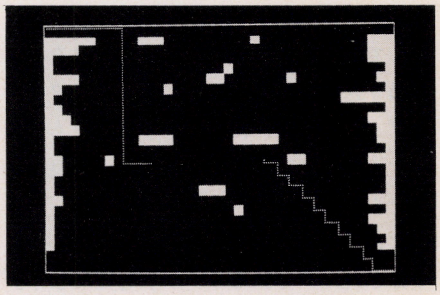
左の人……[A]，[W]，[S]，[Z]

リプレイは[5]のキーです。

プログラムは，それぞれのドットのコントロールがリアルタイムにできるように，INKEY＄で受けた変数はさらに専用の変数に受け渡されるようになっています。

編集部記：この手のゲーム元祖は，確かゲームセンタのTVゲームだったと思います。動作は単純なので割と短いプログラムになっています。友達同士で勝ち抜き戦を行えば，かなり興奮しますね。

でも最初のパターン作りにもう少しアイデアが欲しかったところです。これではパ



ターンがつまらなくなり過ぎます。たとえば横方向ばかりでなく，縦方向にも消えていくとか……。

生き残りゲームリスト

©1983 Y. SHIOMI

```
1000 RANDOMIZE TIME/4:DEFINT A-Z:WIDTH 40,25:CONSOLE 0,25,0,0
1010 INPUT "  ﾐｷﾞｶﾗ  START ｽﾙﾋﾄ ﾉ、ﾅﾏｴﾊ";N1$:PRINT
1020 INPUT "  ﾋﾀﾞﾘｶﾗ START ｽﾙﾋﾄ ﾉ、ﾅﾏｴﾊ";N2$:PRINT
1030 CLS:XN=638:YN=198:X1=-4:Y1=0:XM=2:YM=2:X2=4:Y2=0:A$="4":B$="S"
1040 LINE(0,0)-(639,199),PSET,1,B
1050 LINE(0,10)-(639,180),PSET,1,BF
1060 FOR I=0 TO 500:LOCATE RND*34+1,RND*22+1:PRINT"      ";:NEXT I
1070 GOSUB 1170:IF A$="4" THEN X1=-4:Y1=0 ELSE IF A$="6" THEN X1=4:Y1=0 ELSE IF
A$="2" THEN X1=0:Y1=2 ELSE IF A$="8" THEN X1=0:Y1=-2
1080 XN=XN+X1:YN=YN+Y1:F1=POINT(XN,YN):IF F1 THEN F3=F3+1:GOTO 1110 ELSE PSET(XN
,YN,5)
1090 GOSUB 1170:IF B$="A" THEN X2=-4:Y2=0 ELSE IF B$="S" THEN X2=4:Y2=0 ELSE IF
B$="W" THEN X2=0:Y2=-2 ELSE IF B$="Z" THEN X2=0:Y2=2
1100 XM=XM+X2:YM=YM+Y2:F2=POINT(XM,YM):IF F2 THEN F4=F4+1:GOTO 1110 ELSE PSET(XM
,YM,4):GOTO 1070
1110 PLAY "T220V15S0M2400O7E7C4"
1120 FOR J=0 TO 30:PSET(XM,YM,7):FOR I=0 TO 100:NEXT I:PSET(XM,YM,0):FOR I=0 TO
100:NEXT I:NEXT J
1130 IF F1=-1 THEN SYMBOL(50,80),N2$+"ﾉ､ｶﾁ!",5,2 ELSE SYMBOL(50,80),N1$+"ﾉ､ｶﾁ!",
5,2
1140 LINE(200,170)-(550,195),PSET,0,BF
1150 SYMBOL(210,180),STR$(F3)+" : "+STR$(F4),4,2:LOCATE 5,15:PRINT "PUSH [ 5 ] K
EY TO START"
1160 F1=0:F2=0:S$=INPUT$(1):IF S$="5" THEN 1030 ELSE 1160
1170 S$=INKEY$:IF S$<>"" THEN IF S$="8" OR S$="2" OR S$="4" OR S$="6" THEN A$=S$
 ELSE B$=S$

1180 RETURN
```

[11]は下記のように変更（1060行）

[11]トル（1110行 V15S0M2400）

[8] BEEP（1110行）

11用変更リスト

```
1060 FORI=0 TO 500:SYMBOL(16*INT(RND*34+1),8*INT(RND*22+1)),"■■■■",2,1,0:NEXT I
```

ボイスダンプ ——— 井浦　芳男（新潟県）　for FM-7,8,11

声を登録して，声でマシン語のダンプをします。 RUNすると，声の登録(1)かテープまたはディスクから声データのロード(2)かパス(0)かを聞いてきますから，登録ならば，図のように接続し，1を押してください。まず，0を発音すると，すぐに登録された声がスピーカから出るので，よければリターンキーを，悪ければ，N⓪を押して再度発音してください。これをFまで繰り返します。登録された声データをテープ（ディスク）にセーブすることもできます。登録する声は，女性のほうが良いようです。

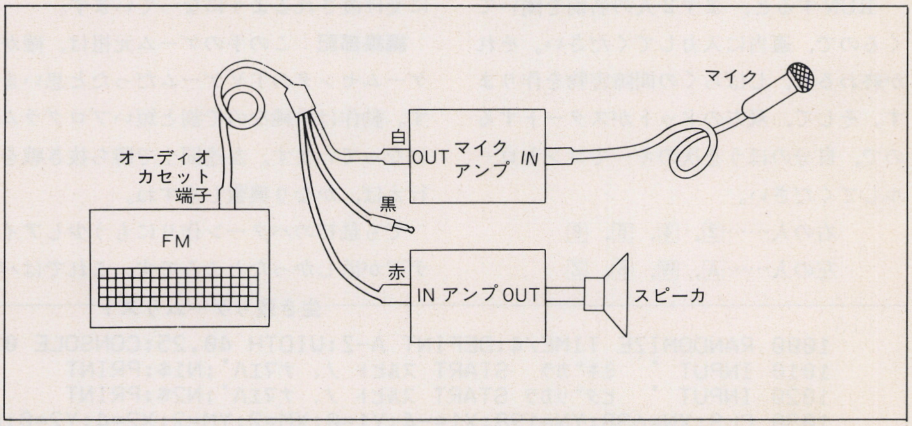
ダンプするアドレスを16進で入力すると，登録された声でメモリダンプを始めます。

このプログラムは，少しの変更でゲームなどにも応用できると思います。変わった使い方としては，ランダムなマシン語部分を読ませると，数分で安らかな眠りに就けます。(参考文献 I/O '81年5月号)

編集部記：来ました，来ました，我ら根暗マイコニストの強〜い味方が。我ら孤独なマイコニストには，打ち込んだマシン語をチェックするとき，読み上げを手伝ってくれる人などいるはずもありません。このプログラムは，打ち込み時に読み上げをしてくれるなんてことはできませんが，チェック時には便利です。しかも30行です！類似品は多くありますが，ここまでシンプルなものは少ないのではと思います。おまけにカセットI/Fを使うのだから，当然データは1か0の1ビット。それでどうしてこんな音が出るのか不思議議です。

発音の登録はなかなか手間でしょうが，一度うまく入ればよいので，問題にはならないでしょう。なお，BとD，CとEが区別できるよう工夫しましょう。井浦さんはA＝アルファ，B＝ブラボ，C＝チャーリ，D＝デルタ，E＝エコーとして登録していました。

かくして，我ら孤独なマイコニストのマシン語入力チェックは，妖気漂うものとなるのだった(はたから見ると無気味〜！)。



ボイスダンプリスト（FM-8の方は色文字のように変更してください）

©1983 Y.IURA

```
100  ***  VOICE DUMP  ***
110 CLEAR &H17FF:DIM A%(30),B%(30)
120    *  ﾏｼﾝｺﾞ READ  *
130 FOR I=0 TO 30:READ A$:A=VAL("&H"+A$):IF A>&H7FFF THEN A=A-&HFFFF-1
140 A%(I)=A:NEXT:RESTORE 190
150 FOR I=0 TO 30:READ A$:A=VAL("&H"+A$):IF A>&H7FFF THEN A=A-&HFFFF-1
160 B%(I)=A:NEXT
170 DATA 343F,108E,1800,8E02,00B6,FD02,8480,27F9,B6FD,0284,8026,F9C6,08B6,FD02
180 DATA 4866,A486,204A,26FD,5A26,F231,2130,1F26,EA35,3F39,,,,,,,,,,,,,,,,,,
190 DATA 343F,108E,1800,8E02,00C6,08A6,A4A7,8C19,4F64,8C15,49B7,FD00,8620,4A26
200 DATA FD5A,26F0,3121,301F,26E3,353F,3900,,,,,,,,,,,,,,,,,,
210    *  VOICE ﾄｳﾛｸ  *
220 INPUT"ﾊﾂｵﾝ ﾄｳﾛｸ(1) LOADM(2) PASS(0)";A:ON(A)GOTO 230,280:GOTO 300
230 FOR A=0 TO 15:GOSUB 370:NEXT
240 FOR A=0 TO 15:GOSUB 390:NEXT:INPUT"ﾅﾆｶ ﾅｵｼﾏｽｶ OK-RETURN NG-(0-F)";A$
250 A=VAL("&H"+A$):IF"0"=<A$AND A<16 THEN GOSUB 370:GOTO 240
260 INPUT"Voice Data ｦ SAVE ｼﾏｽｶ(Y/N)";A$
270 IF A$="Y" THEN SAVEM"VOICE",&H1800,&H3800,&H1800:GOTO 300 ELSE GOTO 300
280 LOADM"VOICE":GOTO300
290 '  *  VOICE DUMP  *
300 INPUT"ｱﾄﾞﾚｽ from";A$:A=VAL("&H"+A$):INPUT"ｱﾄﾞﾚｽ  to ";A$:B=VAL("&H"+A$)
```

（FM-8用変更：170行の1800→10，180行の20→10，190行の20→10）

```
310 FOR II=A TO B STEP 8:SUM=0:AA=II¥256:GOSUB 340:AA=II MOD256:GOSUB 350
320 FOR I=II TO II+7:AA=PEEK(I):GOSUB 340:SUM=SUM+AA:NEXT
330 BEEP1:BEEP0:PRINT":";:AA=SUM MOD 256:GOSUB 340:BEEP:PRINT:NEXT:GOTO300
340 PRINT" ";:FOR T=1 TO 150:NEXT
350 A=AA¥16:GOSUB360:A=AA MOD 16:GOSUB 360:RETURN
360 PRINT HEX$(A);:GOSUB 390:RETURN
370 PRINT HEX$(A);" ｦ ﾊﾂｵﾝ ｼﾃｸﾀﾞｻｲ":A%(2)=&H1800+A*&H200:EXEC VARPTR(A%(0))
380 GOSUB390:INPUT"OK(Y/N)";A$:IFA$="N"GOTO 370:ELSE RETURN
390 B%(2)=&H1800+A*&H200:EXEC VARPTR(B%(0)):RETURN
```

高速セーブ／ロードプログラム——石川　嘉信（名古屋市）　for FM-7

FM-7のカセットI/Fは，平均1600ボーですが，これを2400ボーにスピードアップするプログラムです。

このプログラムを入力したら，カセットにセーブしてください。次にRUNし，カチンという音がしたら，こんどはBREAKキーを押します。FM-7本体後ろのディプスイッチ4番を上げて，プログラムをNEWし，2400ボーにしたいプログラムをロードします。4番スイッチを下げてから，はじめにセーブした本プログラムの後に，2400ボーにしたいプログラムをセーブします。すると，今までより高音でセーブしだします。高速でセーブしているからです。これで，高速プログラムソフトの完成です。

さて，高速ロードの仕方は，カセットを巻き戻して，R.“ ⏎と入力すればOKです。マシン語も同じようにできます。

編集部記：FM-7は処理速度がFM-8の倍近くです。それではテープのロード／セーブも倍近くかというと，互換性など困る点がでてくるので同じにしてあります。でもやろうと思えばできるはずだ，確かにそうです。しかしどうすれば……と思ったらショートプログラムでやってくれた人がいました。もっともほとんどマシン語ですが。

便利だからといって，FM-8ユーザーの友人に渡すテープにこれでセーブしたりしてはダメ。君のテープは読み取れないというレッテルを貼られるから。

念のため付け加えますが，このプログラムを一度実行すると（彼の説明どおり作ったテープをロードしても同じです），LOAD, SAVEは倍速クロックではいつも倍速です。だから，倍速セーブしたテープは，このプログラムを介さずにロードできますし，倍速セーブも単なるセーブで行なえます。しかし，標準セーブのテープを読み出すにはクロックを切り換えなければなりません（そのままでは読み飛ばします）。

高速セーブ／ロードプログラムリスト

©1983 Y.ISHIKAWA

```
10 'BUP PROGRAM
20 FOR BU=&HFC00 TO &HFC36
30 READ BU$:POKE BU,VAL("&H"+BU$)
40 NEXT
50 EXEC&HFC00
60 CLS:LOCATE10,10:PRINT"ﾀﾀﾞｲﾏ ｺｳｿｸ (2400 ﾎﾞｰ) ﾃﾞ LOAD ｼﾃｲﾏｽ｡"
70 LOCATE10,12:PRINT"     ｼﾊﾞﾗｸ ｵﾏﾁｸﾀﾞｻｲ"
80 RUN"
90 DATA 8E,80,00,F6,FD,0F,A6,84,F7,FD,0F,A7,80,8C,FC,80
100 DATA 26,F1,CC,12,12,FD,F2,B3,FD,F2,B5,FD,F3,35,FD,F3
110 DATA 37,CC,7F,EA,FD,02,17,7E,8F,D9,F6,FD,0F,CC,7F,C0
120 DATA FD,02,17,7E,8F,D9,00,00
```

マシン語プログラムをロードする場合は下記のように変更してください。

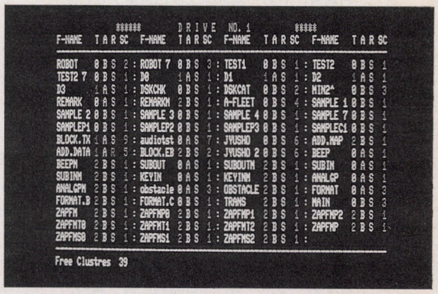
```
80 LOADM"CAS0:",,R
```

FILES強化プログラム——相島　俊雄（東京都）　for FM-7,8,11

FM-7，8のFILESは，単色のうえ，縦書きで不便です。そこで，これを多色横書き表示させ，好きなプログラムを自動的にロード，実行させるプログラムを作りました。

本プログラムをRUNさせると，WHAT DRIVE No.と聞いてきますから，希望のドライブ番号を入力してください。ただしこのときは数字のみでリターンキーは押さないでください。すると，横4個ずつディレクトリリストが表示されます。ディレクトリリストの書式はF-BASICと同じですが，それぞれ色分けされて出力されます。出力が終わると，FILE SELECTの表示とともに，最初のファイル名が緑色で表示されます。カーソル移動キーによって，実行させたいファイル名を緑色にし，リターンキーを押してください。自動的にロード，RUN（あるいはEXEC）してくれます。もし，他のディスケットの内容が知りたいときは，リターンキーではなく，ESC キーを押してください。WHAT DRIVE No.の状態に戻ります。

なお，本プログラム名をSTART UPでセーブし，システムをオートスタートにするとより便利です。



最後に，本プログラムでは，最大76個のファイルしか表示できません。

編集部記：FATとかディレクトリがどうなっているかわかってくると，必ずやるのがこれですね。ロードするファイル名を数

字で指定する，というのはよく作られます。とくに，F-BASIC V1.0～3.0はFILESが1行に1個しか出ないので，すぐ画面からあふれてしまいます。見やすく並べたくなるのは人情です。当然，この手のプログラムの投稿も多く，今回で3つ目です。しかし，カーソルキーによるファイル名選択というすばらしい発想が功をなし，本プログラムが選ばれたわけです。オールBASICながら処理速度もまあまあですし,STARTUPにはちょうどいいでしょう。

しかし，なんといっても目立つのは表示の色です。趣味のあう人はこのまま，趣味の違う人はやはり改造して使うのが一番でしょう。

FILES強化プログラムリスト

©1983 T. AIJIMA

```
10 '**********************************************
20 '*  FM-7 DISK FILES  By T.AISHIMA & K.ENDO    *
30 '*  Ver,1   1983 年  11 月  12 日             *
40 '**********************************************
50 CLEAR 2000:CLS
60 DIM ND$(152),ML(152)
70 PRINT "WHAT DRIVE NO = ";:
80 DRV=VAL(INPUT$(1)):PRINT DRV:IF DRV>1 THEN 70
90 CONSOLE0,20,0,0:CLS:WIDTH80,25
100 COLOR3:PRINT"                ******            D R I V E   NO.";DRV;"              ***
**"
110 COLOR6:FOR I=1 TO 4:PRINT "F-NAME   T A R SC   ";:NEXT I:COLOR7
120 FOR I=1 TO 77:PRINT "-";:NEXT I
130 ND=0:L=0:CS$=DSKI$(DRV,1,1):FOR J=4 TO 32:DIR$=DSKI$(DRV,1,J)
140 KK=0:FOR K=1 TO 225 STEP 32:KK=KK+1:D$=MID$(DIR$,K,8):GOSUB 460
150  IF ASC(D$)=255 THEN K=225:J=32:GOTO 280
160  IF ASC(D$)<>0 THEN D$=LEFT$(D$+"        ",8):PRINT D$; ELSE 280
170 ND$(ND)=D$
180  FT=ASC(MID$(DIR$,K+11,1)):FT$=STR$(FT):ML(ND)=FT:ND=ND+1
190 IF FT=0 THEN FC=5
200 IF FT=1 THEN FC=1
210 IF FT=2 THEN FC=2
220  AF=ASC(MID$(DIR$,K+12,1))
230  RF=ASC(MID$(DIR$,K+13,1))
240  IF AF=&HFF THEN AF$="A":AC=4 ELSE AF$="B":AC=6
250  IF RF=&HFF THEN RF$="R":RC=1 ELSE RF$="S":RC=3
260 L=L+1:IF L=9 THEN L=1
270 COLOR FC:PRINT FT$+" ";:COLOR AC:PRINT AF$+" ";:COLOR RC:PRINT RF$+" ";:COLO
R 2:PRINT SC$;:COLOR 7:IF L=4 OR L=8 THEN PRINT "    ";:GOTO 280 ELSE PRINT " : "
;
280 NEXT K,J
290 Z=L/4:IF Z=1 OR Z=2 THEN GOTO 300 ELSE PRINT
300 FOR I=1 TO 77:PRINT "-";:NEXT I
310 PRINT "Free Clustres",DSKF(DRV)
320 ND=ND-1
330 X=0:Y=3:XX=0:YY=3:F=0
340 LOCATE 0,23:PRINT "FILE SELECT";
350 LOCATE X,Y:COLOR 4:PRINT ND$(F);:COLOR 7:GOTO 370
360 LOCATE XX,YY:PRINT ND$(FF);:XX=X:YY=Y:GOTO 350
370 KD$=INPUT$(1)
380 IF KD$=CHR$(&H1E) THEN IF F-4<0 THEN BEEP:GOTO 370 ELSE FF=F:YY=Y:F=F-4:Y=Y-
1:GOTO 360
390 IF KD$=CHR$(&H1F) THEN IF F+4>ND THEN BEEP:GOTO 370 ELSE FF=F:YY=Y:F=F+4:Y=Y
+1:GOTO 360
400 IF KD$=CHR$(&H1C) THEN IF F+1>ND THEN BEEP:GOTO 370 ELSE FF=F:XX=X:YY=Y:F=F+
1:X=X+20:IF X>79 THEN XX=60:YY=Y:X=0:Y=Y+1:GOTO 360 ELSE 360
410 IF KD$=CHR$(&H1D) THEN IF F-1<0 THEN BEEP:GOTO 370 ELSE FF=F:XX=X:YY=Y:F=F-1
:X=X-20:IF X<0 THEN XX=0:YY=Y:X=60:Y=Y-1:GOTO 360 ELSE 360
420 IF KD$<>CHR$(13) THEN450
430 IF ML(F)=0 THEN RN$=HEX$(DRV)+":"+ND$(F):CLS:LOAD RN$,R:END
440 IF ML(F)=2 THEN RN$=HEX$(DRV)+":"+ND$(F):CLS:LOADM RN$,,R:END
450 IF KD$=CHR$(&H1B) THEN CLS:RUN ELSE BEEP:GOTO 370
460 SC=ASC(MID$(DIR$,K+14,1)):CT=0
470 IF SC>=&HC0 AND SC=<&HC7 THEN 510
480 CT=CT+1:IF SC>&HC7 THEN 530
490 SC=ASC(MID$(CS$,SC+6,1))
500 GOTO 470
510 SC$=STR$(CT)
520 SC$=RIGHT$("  "+SC$,2)
530 RETURN
```

漢字拡大表示プログラム ――成瀬 健一(国分寺市)　for FM-7,8,11

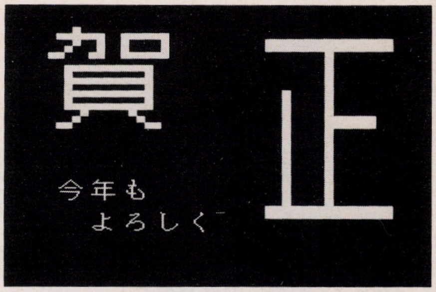
漢字ROMより得られる文字および記号を縦・横任意の整数倍に拡大し，画面に表示するプログラムです。

使い方は，Gカーソルを漢字を表示したい位置の左上の角にもっていき，リターンキーを押します。下方にその位置の座標の値が表われますから，確認してから，漢字コードを16進数で入力してください。何も入力しない場合は，取り消しとなり，Gカーソルの移動に戻ります。

次に，横の倍率,縦の倍率を入力します。拡大すべき漢字と，拡大したときの大きさを示す四角が点滅されますので，間違いがなければYを押し，取り消す場合はNを押してください。

なお,ハードコピーをとる場合は，PF10を押し，リターンキーを押してください。

編集部記：拡大の仕方に新鮮味を感じ，趣味で選ばせていただきました。実用になるかどうかは別として，このようなプログラムテクニックもある，ということで，このプログラムを採用させていただきました。



大いに参考にしてほしいと思います。Y/Nの入力時，カナモードになっていても大丈夫というのもにくいですね。

漢字拡大表示プログラムリスト

©1983 K. NARUSHE

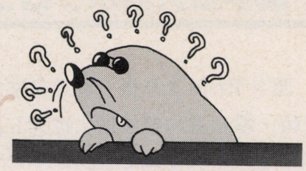
```
100 WIDTH 40,25:CONSOLE 0,24:CLS 3:DEFINT A-C
110 DIM A(199):DIM B(319):DIM C(199):D$="NnﾐYyﾝ"
120 ON ERROR GOTO 400:ON KEY(10) GOSUB 390:KEY(10) ON
130 COLOR 2:GCURSOR(P,S),(P,S):COLOR 7
140 GET@(0,192)-(639,199),B,G
150 PRINT"(";P;",";S;") ";:INPUT"ｶﾝｼﾞ ｺｰﾄﾞ";K$:IF K$="" THEN GOSUB 380:GOTO 130
160 K=VAL("&H"+K$)
170 INPUT"ﾖｺﾊﾞｲﾘﾂ";J:IF P+J*15>639 THEN BEEP:GOTO 170
180 INPUT"ﾀﾃﾊﾞｲﾘﾂ";I:IF S+I*16>199 THEN BEEP:GOTO 180
190 GET@(P,S)-(P+15,S+I*16-1),C,G:LINE(P,S)-(P+15,S+I*16-1),PRESET,,BF
200 PRINT@(P,S),K:GET@(P,S)-(P+15,S+15),A,G:PRINT"(Y/N)";
210 FOR W=0 TO 1:GOSUB 370:FOR V=0 TO 80:C$=INKEY$:IF C$=""THEN NEXT:NEXT:GOTO 2
10
220 IF W-1 THEN GOSUB 370
230 IF INSTR(D$,C$)=0 THEN 210 ELSE GOSUB 380
240 IF INSTR(D$,C$)<4 THEN 350
250 FOR Q=15 TO 0 STEP -1:FOR R=0 TO I-1
260 A(Q*I+R)=A(Q)
270 NEXT:NEXT
280 LINE(P,S)-(P+15,S+15),PRESET,,BF
290 PUT@(P,S)-(P+15,S+I*16-1),A,PSET
300 FOR Q=14 TO 0 STEP -1
310 GET@(P+Q,S)-(P+Q,S+I*16-1),A,G:LINE(P+Q,S)-(P+Q,S+I*16-1),PRESET
320 FOR R=0 TO J-1
330 PUT@(P+Q*J+R,S)-(P+Q*J+R,S+I*16-1),A,PSET
340 NEXT:NEXT:GOTO 360
350 LINE(P,S)-(P+15,S+15),PRESET,,BF
360 PUT@(P,S)-(P+15,S+I*16-1),C,PSET:GOTO 130
370 LINE(P,S)-(P+J*15-1,S+I*16-1),XOR,,BF:RETURN
380 CLS 3:PUT@(0,192)-(639,199),B,PSET:RETURN
390 HARDC 2:RETURN 130
400 RESUME NEXT
```

（行180の「199」について）11 の400ラインのときのみ399に変更

ショートプログラム応募要領

- ■1行80字以内で，できるだけ30行に収めてください。
- ■きれいなプログラムを望みます。マルチステートメントは多用しないでください。
- ■プログラムは，カセットテープまたはディスケットにいれてお送りください。ディスケットの場合は，新しいディスケットをお返し致します。
- ■プログラムの説明，使い方など必要なことを簡潔にまとめてください。
- ■プログラムおよび原稿は，多少修正させていただく場合があります。
- ■他誌との二重投稿，盗作など，モラルに反する応募は固くお断わり致します。
- ■応募原稿は原則としてお返し致しません。
- ■応募された方全員に，Oh！FMシールとシャープペンシルをお送り致します。
- ■すぐれた作品は本誌に掲載し，原稿料にかえてOh！FM特製Tシャツと図書券5,000円分をお送り致します。
- ■原稿にも，住所・氏名・年齢・職業などを明記して，〒102 東京都千代田区四番町2－1 ㈱日本ソフトバンクOh！FMショートプログラム集係までお送りください。

Question & Answer



Q FM-7本体のスピーカ端子とラジカセのマイク端子をつないだのですが，全く鳴りません。どうしたらよいのでしょうか。（千葉県・斉藤喜之）

A そうですか，鳴りませんか。FM-7とラジカセのつなぎ方は，ラジカセの説明書には載っていないでしょうから，この際，説明しましょう。

マイク端子に接続して音をラジカセのスピーカから鳴らすというのはたいていのラジカセではできないのです。というのは，マイクをつないだとき，拾った音をスピーカから出すと，ハウリングといってヒーッという音が出る現象を生じやすいのです。そのため，マイク入力の音は録音状態にしてイヤホン等を使わないと聞けません。富士通のデータレコーダのように録音中の音が聞こえるものや，マイクミキシングのできるものを除き，マイク入力をそのままスピーカに出すことはできないと思ってください。

ラジカセには，LINE INやAUX IN等の入力があるものが多くあります。これらの端子を使えば，FM-7の音をラジカセのスピーカから出力することができます。自分のラジカセの説明書をめくってみてください。

Q 初心者です。パソコン用語でアセンブラというのはわかるんですが，マクロアセンブラとかクロスアセンブラというのは何ですか。教えてください。（大阪府・柳沼泰教）

A アセンブラを実行するコンピュータとそのオブジェクトを実行するコンピュータは，普通は同じですね。このようなアセンブラをセルフアセンブラと言います。しかし，アセンブルするコンピュータとオブジェクトプログラムを実行するコンピュータが違うものがあってもいいわけです。これがクロスアセンブラです。たとえば6809の機械語で，8086のプログラムを作れれば，それは6809用8086クロスアセンブラというわけです。

次にマクロアセンブラですが，これは普通のアセンブラにマクロ機能といって，いくつかの命令をひとまとめにして新たに命令（アセンブラ上での）を作れるものです。たとえば，ASLB, ROLAの２命令をASLDという命令として定義すれば，後はASLDとするだけで，機械語になるときにASLB, ROLAの２つ分になります。またORCC ＃801をCLCと定義すれば，ソースリストはずっと見やすくなります。このように，マクロアセンブラは，普通のアセンブラよりもプログラムの開発がずっと楽になるものです。

Q 6809の命令一覧表で，インヘレント，ダイレクト，エクステンド，イミディエイト，インデックス①，リラティブ等と表の上にありますが，何を意味しているのでしょうか。（秋田市・寛太郎）

A これらはアドレッシングモードを表しています。アドレッシングモードというのは，命令の対象となるメモリを指定する様式のことです。インヘレントは命令中で操作の対象が指定されているもの，イミディエイトとはオペコードの次の１バイト（または２バイト）の値，ダイレクトは上位８ビットがＤＰレジスタの値で下位８ビットがオペコードの次の１バイトで指定されるアドレスの内容，エクステンドはオペコードの次の２バイトで示されるアドレスの内容，インデックスはさらに細かく分かれており，オペコードの次にくるポストバイトというものがアドレッシングモードを指定します。なお，命令表に付いている①は，注の①を見よということです。

命令はアドレッシングモードによってコードが異なります。ある命令のオペコードを調べるときは，その命令の行の目的のアドレッシングモードの欄を見るわけです。

例を挙げると，INCAというのは，アキュムレータが対象で，これはオペコード中に対象を含むのでインヘレント（アキュムレータアドレッシングとするものもある），INC　$2000というのはエクステンドとなるので，オペコードを調べるときは，INCのうちエクステンドの欄を見ます。また，INC，Xはインデックスなので，インデックスの欄を調べ，次いでポストバイトをポストバイト表で調べる，という手順で機械語のコードを調べます。

Q FM-7を使って時刻表示やタイマとしても使いたいのですが，FM-7を使っているとだいぶ過熱してくるので，１日中スイッチをオンにしておくと，本体に悪影響を及ぼすのではないかと思い，使用を控えています。FM-7は電源をつけっぱなしにしておいてもいいのでしょうか。

（東京都・高原大輔）

A FM-7は電源入れっ放しだと熱くなる度合が高い部類に入るパソコンです。しかし熱くなるといっても火傷するほど熱くなるということはないはずです。温度は電源を入れたときから同じペースで上がり続けるわけではありません。ある温度でそれ以上は高くならないものです。だから，１日中つけっ放しでも大丈夫です。ただし，風通しの悪い所で夏の暑い日につけっ放しにするのは少々不安が残るので，こういう環境で長時間放置するのはやめたほうがよいでしょう。

＊　＊　＊

ＦＭに関する質問をお寄せください。質問内容は文書で，できるだけ具体的にお書きください。ソフト・ハードなど幅広い質問をお待ちしています。ただし，マニュアルを読めばすぐわかるような質問はご遠慮ください。住所・氏名・電話番号などを明記のうえ，Oh！ＦＭ編集部Ｑ＆Ａあてお送りください。

なお，質問に対する回答は誌上のみです。往復はがき等による質問はご遠慮ください。

郵便はがき

料金受取人払

麹町局
承認
245

差出有効期間
昭和59年4月
19日まで

102

（受取人）
東京都千代田区四番町二―一

株式会社　日本ソフトバンク
出版部行

読者の皆さまのご意見をお寄せください。誌面づくりに反映させるための貴重な資料となりますので、ご協力ください。

愛読者アンケート

○本誌をご覧になった感想　　　　　　プレゼントNo.

○いちばん良かった記事

○興味のなかった記事

○載せてほしい記事・内容を具体的に

○どんなパソコンをお持ちですか　（個人用　会社等　ない）

本体（　　　　　　　　）　購入　年　月

CRT〈グリーン・カラー〉（メーカー名　　　　　　）／家庭用

Dレコーダ（メーカー名　　　　　　）／ラジカセ

ディスク〈S・D〉（型番 メーカー名　　　　　　）

プリンタ（型番 メーカー名　　　　　　）

漢字ROM　Z-80カード　RS-232C　PSG　バブル

その他（　　　　　　）

○今，一番気にいっている市販ソフト（発売元も）

○今後Oh!シリーズとして新たに取り上げてほしいハードは？

APPLE SMC MULTI FP MB SORD JR MSX その他（　　　）

氏名	年齢	性別 男　女	マイコン歴 年　か月
住所 〒 Tel			
職業(勤務先，部署，役職) ※			

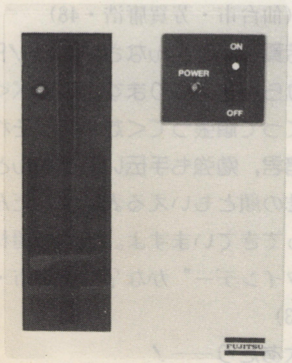
FM　第7号　　　※学生の場合，学校学部学年を記入してください。

読者の皆さまのご意見をお寄せください。誌面づくりに反映させるための貴重な資料となりますので、ご協力ください。

プレゼント情報

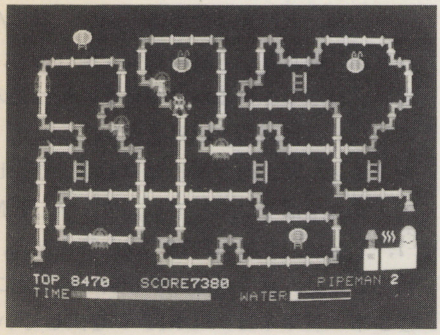
今号のプレゼント商品を紹介します。ご希望の方はとじ込みのアンケートハガキに必要事項記入のうえ，プレゼントナンバーを記入してください。締め切りは2月15日(必着)。当選者発表は4月号（3月18日発売）誌上で行います。

①薄形シングルミニフロッピディスクユニット(MB27607，FM-7あるいはFM-8用)　1名
富士通提供
FM-7の場合はインタフェースカード，ケーブル，システムディスク(V3.0)，FM-8の場合はアダプタ，ケーブル，システムディスク（V2.0）が付属します。



M B27607 薄形ミニ

②PIPE MAN（FM-7，8用）　5名
システム・イン・猫提供



③大阪パソコンFM祭りゲストサイン色紙　6名
月亭八方，林家小染ら有名落語家4名のサイン色紙。

④Oh!FM Vol.7 ノート　10名
本誌今号と同じ体裁・表紙で，中は真白なノートです。

⑤Oh!FMシャープペンシル　50名
Oh!FMの名入りシャープペンシル。

「6809機械語開発システム」モニタ募集！

(株)セルソフトウェアでは，F-BASIC上で動作する「6809機械語開発システム」を開発，発売にあたり，Oh!FM読者より10名をモニタとして募集します。

このシステムは6809CPUのためのアセンブラ，エディタ，シンボリックデバッガを軸としてファイル内容の表示，プリント，ディレクトリ表示などのユーティリティを含む本格的なものです。しかも，FLEXやOS-9などの特別なOSを必要とせず，ミニディスクベースのF-BASIC上で動作するので，BASICと機械語のプログラムとを簡単に組み合せることが可能となります。モニタ希望の方は下記応募要領でどうぞ。

募集対象　FMシリーズ所有者でミニディスク装置を所持，または購入予定の方
募集人数　10名
応募方法　官製ハガキに住所，氏名，年齢，職業，電話番号，所有しているFMシリーズの機種名を書いて下記へ。締め切りは1月31日。

〒102　東京都千代田区飯田橋1-9-6
朝日飯田橋マンション1001号
(株)セルソフトウェア

なお，販売価格は12,800円(送料着払い)です。購入希望者は使用機種名明記のうえ，上記あて現金書留で。購入目安資料は送料込1,000円(機種名，"マニュアル"と明記)。

6号プレゼント当選者

①4人ポーカー（7名）
塚原仁(横浜市)・奈良富士明(秋田市)・水谷雅晴(杉並区)・白石陽介(草加市)・小路和明(茨木市)・和田肇(横浜市)・奥田啓生(吹田市)

②倉庫番（10名）
日高聡（町田市）・鈴木武明（銚子市）・久松達央(新治郡)・松居健一(生駒市)・山田崇生(北九州市)・林佳樹(市川市)・重富憲治（徳山市）・竹ノ内典子（和泉市）・吉田裕史（天理市）・小倉芳昭（千葉市）

③漢字ROMカード（5名）
松島昇（伊達市）・本澤宏文（松本市）・山本賢一（京都市）・石川皇（弘前市）・大谷清治（松戸市）

④コンピュータグラフィックパネル（5名）
冨田桂三（東大阪市）・浜田正一（岡山市）・河上一郎（高松市）・大西光洋（新宿区）・小沢一賀（茂原市）

⑤Oh!FM Vol.6 ノート（20名）
高木妙子(逗子市)・松浦勉(佐世保市)・定森豊(大阪市)・小池由紀夫（船橋市）他16名

⑥FILES Oh!FMノート（10名）
片桐顕（世田谷区）・前原広（横浜市）・天羽敬子(阿南市)・木村進一(川崎市)・泉昇（小松島市）・中橋剛（岐阜市）・奥間啓二(豊中市)・木村郁之(西宮市)・広光洋一（小樽市）・大島俊彦（東大阪市）

⑦Oh!FMTシャツ（a，b各10名）
a．松田泰典（横浜市）・竹内庄司（米沢市）・高松平蔵（御所市）・藤本雅寿（高岡市）他6名
b．中崎義晃（門真市）・尾山純一（福岡市）・紺谷透（神戸市）・前川晶彦（木更津市）・伊藤尚史（横浜市）他5名

READER'S AREA

▶数あるマイコン誌の中でもOh!ＦＭは最高ですね。ところで６号の表紙の太った中年男性はサンタクロースのつもりですか？それとも某編集者？（高松市・池田純子・22）

――あれはサンタクロースなのですが，某編集者をモデルにしたのかもしれません。誰とは言いませんが……(あとがこわい)。

▶ディスク試用記，ユーティリティを載せてほしい。（岡崎市・磯部弘・47）

――特集「ディスク活用術」いかがでしたか？ ディスクもだんだん値段が下がってきているし，ディスクベースのソフトも増えてきています。買って損はない!!

▶「だんだんよくなる法華の太鼓」どおり内容も充実してきたし，バラエティに富んで結構です。６号の表紙，きっとあのサンタクロースの袋の中には，名古屋にやってくるコアラが入っているんですよ。偉大なる田舎，名古屋のＦＭファンはなぜかタモリが憎めなくなってしまった。では最後に名古屋弁で。12月はボーナスシーズン，ＦＭ機がぎょうさん（たくさん）売れて，Oh!ＦＭの読者もびゃあそう（倍増)。ティヤーテー（たいてい）２月からは月刊にしな，いかんかもしれんぜーも。寒なったでみんなかゼひかんよう，あんばよう（あんばいよく）やってちょうよ。（名古屋市・安田正邦・38）

――誰にでもしゃべれる名古屋弁「エビフリャー（エビフライ）」

▶もぐらの名前は「黒メガネモグラ」がいいと思う。（茅ヶ崎市・吉原正・16）

▶もぐらの名前は，メカっぽいもぐらとメガネをかけているところから「メグモグ」がいい。（習志野市・関根賢一・11）

▶これからは初心者のページと上級者のページを作ってみては？ ところでもぐらの名前は「モグ君」がいいと思う。（宇都宮市・栗田弘司・15）

▶あっ！ もぐらの名前は「ねぐら」これっきゃないっ！（堺市・棚次真由美・18）

▶モグラの名前。6809から「マルル」，ＦＭからは「フムフム」。（豊中市・宇土康一・17）

▶もぐらの名前；ふじお（富士男）がいいと思います。（浜松市・関根伸吾・20）

――どうもありがとうございます。しかしいろいろお寄せいただいたのですが，どうも「ナイス！」というのが見あたりません。今度こそは可愛い名前をお願いします。

▶（主人の代筆で書いている子なし結婚７年目の妻です）ＦＭ-7を買って以来，彼女(ＦＭ-7)にかかりっきりなのです。ど～してくれますか。おまけにOh!ＦＭを買ってくるとワキ目もふらずにキーをたたいております。でもまぁいいか……子供みたいなもんでしょう。長～～い目で見守ってかわいがっていきましょうね。ステキなプログラムをつくってくださいな。（金沢市・矢木淑子）

――FM-7をお腹を痛めた子，とでも思えば，ついついかわいくもなるでしょう。それにしてもだんな様，FMばかりかわいがっちゃいけませんよ。

▶友人がパソピア７を買ったので，「Oh!PASOPIA」をすすめました。クリスマスに彼にHC-20をプレゼントするので「Oh!HC」を買いこみました。Oh!シリーズ大好きです。彼はHC-20のかわりに，サーファーギャルの私にはブギーボードを買ってくれます。（名古屋・中村彩子・20）

――のろけもここまでくれば立派なもんだ。しかし，よく書くよ。困ったもんダ。

▶全般に初心者向けでたいへん良かった。我々，中年はパソコンに取り組む時間が少ないので，１冊を実行するのに日数がかかります。したがって隔月刊のままでよい。勉強も家の手伝いもしないでパソコンに夢中になっている若者向けに編集しないように。（仙台市・芳賀庸浩・48）

――芳賀さんごめんなさい。Oh!FMは次号から月刊になります。なるべく時間をつくって頑張ってください。それから若者諸君，勉強も手伝いもちゃんとやろう。

▶雑誌の顔ともいえる表紙が，だんだん良くなってきていますよ。７号の題材は“バレンタインデー”かな？（新潟市・坂井伸之・23）

――大あたり――！

▶とうとうＦＭ-11を買った。お父さん，お母さん，ありがとう。開封の前に身を清め，東西南北に頭を下げ，キリスト，仏様，アラーの神，天照大神にお祈りをし，あらんかぎりの儀式をし，11の雄姿が現れると，思わず目がつぶれるかと顔を手で覆ったのでした。（福岡市・田口茂夫）

▶クイズ。富士通が出したマイクロコンピュータは全部でいくつでしょうか？「ＦＭ-7，ＦＭ-8，ＦＭ-11の３つ」と答えた人，残念でした。富士通はFM-8を出す以前に，ワンボードマイコン“LKIT-8”を出していたのです。これは意外と知られていない事実です！（ななしのごんべえ）

▶いつもお父さんと見ています。ぼくにはまだわからないことが多いです。（横浜市・伊沢正裕・10）

――お父さんと一緒にがんばれ！

▶1年経ったので月刊化してほしい。発売月の次の月の空腹感をどうしろとおっしゃるのですか。ファイル版より月刊化を，月刊化よりも記事を少なくして週刊化を。（春日部市・福田廣士・31）

――てなわけで先述しましたとおり，3月発売分よりOh!FMは月刊になります。長らくお待たせしました。

▶ショートプログラム集に感動した！
ディスアセンブラに狂喜した!!
タイルペイントで天国へ行った!!!
（大和市・長田利明・17）

▶やたらとプログラムにバグが多いのは，次の号（虫つくろいのページ）を買わせるために，わざとやっているのではないか？（大垣市・小川幹夫・19）

――す，すつ，するどい！

▶毎回プレゼントの内容とアンケートを書くのが楽しみでしょうがない？ なんといってもマイコン誌の中では値段が安い。ほっかほっか亭の幕の内弁当を食べるんだったら，一食抜いてOh!FM買います。（竜ヶ崎市・小川光明・25）

――食事を抜いてOh!FMを買ってくださるなんて……。ちなみにほっかほっか亭は地域によってかなり差があるみたいですね。市ケ谷のはあまりよくありません。

▶Oh!FMもたまってきたので本棚へ。あう，本の大きさが！ すなわちOh!FMだけが本棚から飛び出ているのです。だれだ，この本の大きさを決めたのは？（札幌市・河内幸晴・26）

――ハードを使う前にソフトをよく調べよう。

▶突然ですが，僕の悩みを聞いてください。僕の恋人，絵笛・咽文ちゃんとは去年の3月東京で出会い，それからずっとつきあっています。しかしこのごろは，3時間も一緒にいると熱を出し，僕のいうことを理解してくれません。そうです，絵笛・咽文ちゃんはチャタリング病にかかっているのです。それもまだ2ヵ所や3ヵ所ならいいのですが，よくて6ヵ所，最悪のときは9ヵ所という記録も残っています。この病気は富士通病院に入院させれば治ることはわかっていますし，ただだということも知っています。が，噂によると一度入院させると最低でも1ヵ月は入院しなければならないと聞きます。1ヵ月も絵笛・咽文ちゃんの姿を見ないということは，僕にとってとても悲しく，苦しいことです。一体僕はどうすればいいのでしょうか。（新潟県・今成文明・15）

――早期入院，早期治療がいちばん。遠く離れていてこそ，愛が確かめられるというものではありませんか。

▶パソコン歴4ヵ月なので，マシン語のことがよくわかりません。初心者のためにマシン語の詳しいことを教えて！（新潟県・宮下俊章・16）

▶なかなか興味深い。マシン語の初歩をもっと多く解説してほしい。（加古川市・岩泉国弘・31）

――みなさんマシン語，マシン語とおっしゃいますが，日本語の方は大丈夫なんでしょうか？

▶はじめまして。私は去年の10月末にFM-7を買いました。そこでみなさんに質問があります。みなさんはパソコンを買ったら主にどういうことに使用しているのですか？私は買ったのはいいのだけど，まだ何に使用するか迷っています。そこで今後，どういう目的で使用すれば役に立つのか教えてください。お願いしま～～～す。（西宮市・三好真理・20）

▶10月にNECからPC-9801Fが発表されたかと思っていたら，やつぎばやに9801Eが発表された。すでにディスクを持っているユーザーにとって嬉しいEタイプである。富士通も負けずに16ビットパソコンを発表してはどうか。そこで自分なりに新型16ビットパソコンの仕様を考えてみた。まず，CPUはもちろん68000 10MHz バージョン（12MHz でもよい），OSはCP/M-68KかOS-9/68000，BASICはF-BASIC68KかBASIC09,グラフィックコントローラはロジカルイメージコントローラ（LIC・MN8614），メモリは超LSI技術を結集した256KB RAM。ついでに発売は59年2月，価格は218000円。これでいかが!!（一宮市・小川憲男・26）

▶FM-7およびFM-8の記事が多く，FM-11を使用している私としては不満です。11使用の読者もいることを忘れないでほしい。ウルトラ7やエイトマンもよいが，私は人間的なナポレオンソロが好きです。（米沢市・中沢一成・24）

▶プレゼント発表のとき「他○名」というのは絶対やめろー。（岩見沢市・佐藤衛・12）

――ごめんなさい。今回もやってしまいました。でもOh!FM買っとくれ。

▶土浦市・富田剛

Information

アームロボ抽選会

東京・虎ノ門のマイコンスカイラブにて，ユーザー代表小林真人君，地元警察官らの立ち会いのもと，アームロボおよび富士通オリジナルトレーナーの厳正なる抽選が行われた。

抽選は，応募ハガキに通し番号を打ち，抽選機で数字を出す，という方法で行われた。応募総数 8,352 通のなかから，1等アームロボは兵庫県尼崎市の伊藤彰さん，東京都文京区の竹内健二さんはじめ 200 名が，2等のオリジナルトレーナーは愛知県尾西市の岩田隆央さん，埼玉県熊谷市の荻山正生さんはじめ 2,000 名が，それぞれ賞品を勝ち取った。

ウーン，応募数8,352通で，2,200名当選？ 送っておけばよかったなー，というユーザーもかなりいるのでは？ なお，発送は年内とのことですから，まだ何も着かない応募者は，ハズレ。いさぎよく諦めましょう。

'83 大阪パソコンFM祭り

12月19日から21日の3日間，大阪心斎橋プラザビルで，パソコンFM祭りが開催された。これは本誌5号で紹介済みだが，サンシャイン文化会館で行われた「パソコンFM祭り」の大阪版。ただし，FM-Xの発表後だったため，FM-Xが出展されるなど東京とはちょっと違った雰囲気だった。

会場は35の協賛出展会社のブースと，FM

-Xを使ったスロットマシン実演のイベントコーナー，富士通よろず相談所，富士通のソフト・ハードを展示した富士通ブースで構成され，イベントコーナーでは明石家さんまをはじめ，人気のお笑いタレントによる「FMまつり寄席」が開催され，会場に笑いをふりまいた。

オリジナルプログラム募集

①オリジナル投稿プログラム
②FM-7用音楽プログラム

歌謡曲，クラシックなど音楽のジャンルは問いません。ただし自分でFM-7のPSG用にアレンジしたものにかぎります。参考文献，使用曲名，資料など必ず明記してください。

両方とも①プログラムの内容，特徴，アルゴリズムなどの説明，②フローチャート，変数表，③遊び方や操作方法，注意事項，④プログラムの入ったテープないしはディスケット(氏名明記のこと)，⑤機械語の場合はソースプログラム添付，以上を住所・氏名・年齢・職業・電話番号などを明記のうえ，郵送するか直接お持ちください。締め切りは特に設けません。あて先は，

〒102 東京都千代田区四番町2-1
㈱日本ソフトバンク　FM投稿係

- プログラムは未発表のものに限ります。
- 他誌との二重投稿，盗作などモラルに反する応募は固くお断わりします。
- 応募原稿は原則としてお返し致しません。
- 掲載にあたって多少修正させていただくことがあります。
- 応募者全員にOh!FMシャープペンシルを進呈。
- 掲載の場合は本誌規定の原稿料をお支払いいたします。

PIAプリント基板のお知らせ

「はじめて握るはんだごて」のジョイスティック，または次回のタイマを製作したいと思ってもPIA基板を作っていない，作りたくても誤配線が心配だ，という人たちへ朗報。

清瀬市の須田氏より，PIAのプリント基板（ガラスエポキシ，両面スルーホール，シルク印刷付き）を製作した，とのお便りをいただきました。ジャンパ線を使用しての製作に失敗したり，あるいは初めから製作をあきらめてしまった読者に，1枚3,000円（実費）で分けていただけるとのこと。

編集部にて動作確認済み。ただし回路は，TTLが手に入りにくいため，亜土電子にて変更したもの。希望者は2月20日までに直接，ハガキで須田氏に申し込んでください。

〒204 東京都清瀬市松山2-17-2-205
須田哲央氏あて
☎0429-54-6538

Oh!FM月刊に!

長らくお待たせしましたが，3月17日発売分よりOh!FMは月刊になります。よって次号より4月号と号称を改めさせていただきます。発売日は毎月18日です。今後とも引き続きご購読のほど，よろしくお願いいたします。

▶READER'S AREAにご意見，ご感想，ご要望などをお寄せください。内容は問いません。イラストなども歓迎です。掲載者全員にFMステッカーを進呈。

▶売る，買う，仲間募集などはExpressへ。ただしソフトの売買，交換などは扱いません。取り引きに関しては当事者間でお願いします。

〒102 東京都千代田区四番町2-1
㈱日本ソフトバンクFMエリア係

▶4月号は3月17日(土)発売予定です。

(株)日本ソフトバンク発行のパソコン情報誌

月刊

Oh!PC

2月号　発売中

■特集　どかっとたくさん
グラフィックツール
——市販ツールの紹介と
オリジナルツールの公開
■ゲーム大好きディスクユーザーのためのS-DOS80
■PC工作入門
■ゲームをつくるための
機械語サブルーチン集
■機械語，マシン語，真心語
■DISK CHARGE STEP1

月刊

Oh!MZ

2月号　発売中

■特集アクティブ・グラフィクス
MOVING 3Dグラフィクス
作法/COLOR PATTERN
MAKER/LOGOとグラフィクス／超入門スーパーグラフィクスの世界／他
■マシン語ゲームYETI
■ほのぼのシミュレーション
太公望と猫
■記録・再生プログラム
囲碁VTR
■CP/M&F-DOS
■THE SOFTOUCH
TEST/REPORT/NEWS

季刊

Oh!HC

第6号　発売中

■特集　「ビジネス&スポーツ
アプリケーション」
成績処理，電子手帳，飼養計算，土木作業能力計算，テニス・バドミントン成績集計
■グラフィック機能拡張
SGS-sys6(続)
■厳しいチェック！確定申告
■ニューバージョン　QC-10
TYPEⅡの全容
■CP/Mガイダンス(中級編)
■簡易言語と実務(2)

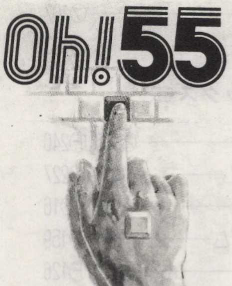
季刊

Oh!PASOPIA

創刊2号　発売中
1984．Winter

■グラフィックス特集
■京都の四季を楽しむ！
■オリジナルアニメに挑戦
■うわさの野球ゲーム，ついに登場
■リズムボックスをパソコンで
■お待たせしました！T-BASIC集
■オリジナルゲーム，MAZYの秘宝／他
■金銭出納帳，リスト公開(OA-BASIC)
■日本語ワープロを使って（パソピア16用）

Oh!55



第2号　近日発売予定

■特集/5550用パッケージソフト
■IBM.PC(5150)で5550の可能性を探る
■逆ポーランド記法実習プログラム
■MS-DOS用スクリーンエディタ
■マクロアセンブラ
■ケーススタディ／5550導入オフィス
■プログラム／バイオリズム相性率ほか

季刊

THE SOFT BANK⑧

1月下旬 発売予定

■特集／簡易汎用ソフト
業務からみた活用法と各ソフトの解説
■新作ワープロ・ソフトを試用する
■'84ソフト最前戦レポート
統合型ソフトの動向を探る
■モニタ報告
市販ソフト試用レポート
■ビジネスソフト機能比較
■●機種別／'84版最新ジャンル別ソフトウエアリスト

(株)日本ソフトバンク発行のパソコン情報誌

INDEX
Oh!FM Vol.1～Vol.7

①～⑦はそれぞれVol.1，Vol.2……Vol.7を，Ⓕは FILES Oh!FMを示します。その後に続く数字はページを示します。

●マシン語解説●

6809マシン語入門 ―― ①59,②97,Ⓕ200
BIOS資料 ―― ①69,Ⓕ252
高速3Dグラフィックス ―― ①106,Ⓕ240
マシン語って何んなのですか？ ―― ③17,Ⓕ192
演算のアルゴリズムとプログラム例 ―― ③25,Ⓕ219
中間色PAINTもどきプログラム ―― ③33,Ⓕ227
サブシステムプログラム転送・実行 ―― ③38,Ⓕ232
FM-8の命令を増やそう ―― ③104,Ⓕ247
目的・予算別・マシン語開発ツール ―― ④18
初心者向けプログラム開発話――外字登録エディタ ―― ④22
ソート＆画面操作 ―― ④37
簡単なサブシステムの駆動法 ―― ④45
マシン語でPSGをコントロールしよう！ ―― ④89
BIOSの使い方 ―― ⑥36
続・BIOSの使い方(1) ―― ⑦46
オールRAMモード活用法 ―― ⑦88
マシン語ゲームの作り方 ―― ⑦97
REM文おもしろテクニック ―― ⑦103
6809機械語の書き方(前) ―― ⑦111

●BASIC構座●

F-BASIC中級入門 ―― ①84,②46,③46,Ⓕ149,⑦65
ソフトウェア設計法入門講座 ―― ①128,②90,③74,Ⓕ132,④117
プログラムの上手なつくり方 ―― ②118,Ⓕ276
ベーシックBASIC ―― ③68,Ⓕ121
初心者のためのF-BASICプログラミングアプローチ ―― ③65,Ⓕ118
ゲームリスト入力のテクニック ―― ③49,Ⓕ56
金種計算のプログラム ―― ①29,Ⓕ127
画面制御から入るF-BASIC入門 ―― △100
グラフィックコマンドを使ってみよう ―― ⑤18
構造化プログラミングの思想 ―― ⑤133
構造化プログラミングの理論 ―― ⑦81
SOUND,PLAY文を使っての音づくり ―― ⑥65

●Logo●

FM Logoってこんなもの ―― ⑥17
君も亀になれ！ ―― ⑥22
FM Logoで人工知能を作ろう！ ―― ⑥30

●ゲーム●

モノポリー ―― ①17,Ⓕ62
ジミーラミー ―― ①105,Ⓕ74
マジックカードテレパシー ―― ①112,Ⓕ106
さがしものゲーム ―― ②67,Ⓕ81
川下り ―― ②73,Ⓕ87
ウェディングレース ―― ②76,Ⓕ89
アニマルミステリー ―― ②120,Ⓕ110
ミサイルコマンダー ―― ③61,Ⓕ300
B-1核爆撃機 ―― ③50,Ⓕ97
アルデバランパート1 ―― ③55,Ⓕ102
スペースシャトル ―― ④81
もぐらたたき ―― ④107
アステロイド ―― ⑤60
スカッシュ ―― ⑤78
スロットマシンプログラム ―― ⑤99
エイリアンFMクラッシュ ―― ⑤118
クラシックゲームブロック崩し ―― ⑥49
4人ポーカー ―― ⑥55
3次元迷路 ―― ⑥130
サイモンゲーム ―― ⑥131
GP-7マシン語ゲームサンプル ―― ⑦72
エイリアンブロックくずし ―― ⑦97
アステロイドフリート ―― ⑦107
生き残りゲーム ―― ⑦123

●グラフィックス●

高速3Dグラフィックス ―― ②106,Ⓕ240
中間色PAINTもどきプログラム ―― ③33,Ⓕ227
グラフィックソフトパッケージ ―― ③12,Ⓕ16
三次元パッケージのアルゴリズム ―― ③81,Ⓕ159
Rainy Day ―― ③114,Ⓕ126
グラフィックコマンドを使ってみよう ―― ⑤18
拡大・縮小機能付きグラフィックエディタ ―― ⑤43
PCGの製作と使い方 ―― ⑤55
Country ―― ⑤78
FM Logo画面写真集 ―― ⑥14
タイルペイントプログラム ―― ⑥93

●音　　楽●

PSGサウンドエディタ ―― ④89
サウンドジェネレータ ―― ⑥73
トッカータとフーガニ短調 ―― ⑥80
PSGとリズムボックスを同期演奏しよう！ ―― ⑥88

●開発ツール●

Oh！FM用チェックサムプログラム ―― ③23,Ⓕ198
6809セルフアセンブラ ―― ④31
機械語版ディスアセンブラ ―― ⑥116
ディスクエディタ「ZAPFM」 ―― ⑦27
FLEX→F-BASICファイルコンバータ ―― ⑦31
ワードサーチ＆リプレイスプログラム ―― ③113,Ⓕ158
F-BASICプログラミングツール ―― ⑤105
ラベルコンバータ ―― ⑥102
使用変数名表示プログラム ―― ⑥129
パターンエディタ ―― ⑤25
拡大・縮小機能付きグラフィックエディタ ―― ⑤43
ミュージックメイカープログラム ―― ⑥70
サウンドジェネレータ ―― ⑥73

●実用プログラムリスト●

金種計算のプログラム ―― ①29,Ⓕ127
タイプ練習プログラム ―― ②33,Ⓕ57
リサージュ波形 ―― Ⓕ165
外字登録エディタ ―― ④22
モールス信号プログラム ―― ④109
リサージュ ―― ④110
グラフ作成ツール ―― ⑤36
漢字データ文作成プログラム ―― ⑤79
スペシャルかけ算 ―― ⑥132
スキューをかけるプログラム ―― ⑦36
ステップレート変更プログラム ―― ⑦42
カタログ表示＆ディスケット再利用プログラム ―― ⑦44
蓉子の住所録 ―― ⑦65
ボイスダンプ ―― ④100
高速セーブ／ロードプログラム ―― ⑦124
FILES強化プログラム ―― ⑦125
漢字拡大表示プログラム ―― ⑦126

●F-BASICの拡張●

FM-8の命令を増やそう ―― ③104,Ⓕ247
タイルペイントプログラム ―― ⑥93
裏RAMコマンドを拡張 ―― ⑦88

●DOS●

OS-9+BASIC09 ①88, Ⓕ168
FLEX入門 ②40, Ⓕ177
CP/M-80とその使い方 ③96, Ⓕ183
OS-9 V1.2とBASIC09 ⑤81

●ソフト紹介●

鉄筋コンクリート製図作成 EASY-DRAW ①100, Ⓕ272
点検・ワープロソフト ①36, Ⓕ268
プログラムの上手なつくり方 ②113, Ⓕ276
グラフィックソフトパッケージ ③12, Ⓕ16
FMCALC―その基本機能と応用 ③92, ④74, Ⓕ283
英文ワープロ「イージー・ライター」試用記 ②118, Ⓕ281
「PCA財務会計」使用記 ④114
「BASICジェネレータ」試用記 ⑤94
F-BASIC86 ⑥113
SOFT REVIEW ⑥120, ⑦116
ゲームプロセッサGP-7試用記 ⑦72

●工　作●

TTL2個でつくる倍速基板 ③59, Ⓕ298
はじめて握るはんだごて
　インタフェースを作ろう ④61
　ジョイスティックの製作 ⑤65
　PIAの使い方 ⑥107
PCGの製作と使い方 ⑤55
マニピュレータの製作 ⑦52

●ハード解析●

FM-7回路図と解説 ②49, ③115, Ⓕ306
FM-7ハード解析 ①54, Ⓕ34

●新　機　種●

新機種FM-7の全容 ①41, Ⓕ22
V3.0のニューコマンド ①46, Ⓕ27
FM-7ハード解析 ①54, Ⓕ34
FM-7のベンチマークテスト ①34, Ⓕ38
徹底比較FM-7, FM-8 ①138, Ⓕ8
本格派パソコンFM-11の全容 ②18, Ⓕ40
Photograph Report FM-11 ②4, Ⓕ12
COMDEX/FALL '82に出品されたFM-16 ②31, Ⓕ52
ニューマシンFM-11登場 ①58
FMシリーズに新しい仲間・FM-X新発売 ⑥99
FM-X続報 ⑦93

●周辺機器●

I/O拡張ユニット ラトックREX-8 ①97, Ⓕ304
各社FDUの比較検討 ①84, Ⓕ292
FP-80, RP-80 ②32
A/D変換カードREX-8040 ②121
グラフィックツールハード編 ⑤33
68008カード(α-CARD)試用記 ⑤114
音声入出力モジュール試用記 ⑤130
小型熱転写プリンタMB27407 ⑥72
FM-7用ひらがなROMカード ⑥79
漢字プリンタMB27410 ⑥114
FMシリーズにつながるフロッピディスクカタログ集 ⑦21

●情　報●

ザ・コンピニオン ②38
コンピュートないと ②30
マイコン利用者認定試験 ③115
マイコンショー'83 ④13
FMシリーズマニュアルバグ情報
　「システム解説」編 ④72
　「システム仕様」編 ⑤92
　「F-BASIC入門」編 ⑥100
　「F-BASIC文法書」編 ⑦51
SIGGRAPH'83にみる世界のCGの現状 ⑤14
第1回「パソコンFM祭」 ⑤97
データショウ'83 ⑥16
FD-7, FM-JWP/7, ミュージックハーモナイザ他 ④44
SUPER TDOS, F漢, RM-6 ⑤54

●読　み　物●

コンピュータについての軽いお話 ③88, Ⓕ114
マシン語って何んなのですか? ③17, Ⓕ192
目的・予算別マシン語開発ツール ④18
8ビットCPU談義 ④55
おもしろパソコン用語集 ④98
ディスク用語集 ⑦24
ディスク使用のメリット ⑦17

●ケーススタディ●

山城デザインスタジオ ③78, Ⓕ287
東工大・吉田研究室 ⑤116

●虫つくろい(訂正欄)●

Vol.1の訂正 ②124
Vol.2の訂正 ③48
Vol.3の訂正 ④60
Vol.4の訂正 ⑤89, ⑥87
Vol.5の訂正 ⑥87, ⑦51
Vol.6の訂正 ⑦51

●特別付録●

6809インストラクションポケットブック ④
レイアウトデザインシート ⑤29
パターンデザインシート ⑤31

バックナンバのお知らせ

Vol.1～3は品切れです。FILES Oh! FM(Vol.1～3の再編集号)をご利用ください。Vol.4～6については若干の在庫がありますので，お近くの書店にお申し込みください。ご注文の際には，下記の雑誌コードを告げていただければより確実です。

Vol.4　雑誌02199-08

Vol.5　雑誌02199-10

Vol.6　雑誌02199-12

FILES　雑誌02200-11

DiskFM

¥3,000(5″ディスケット1枚)

1月18日 Vol.2発売

Disk FMは、富士通FM-7/8/11のディスク・ユーザーのためのマガジンです。
今回は、各種ユーティリティ、高速タイル・ペイントを用いた
アニメーション、シミュレーション・ゲーム等です。

●Disk FMVol.1は1月18日より¥5,000に値上げされました。なにとぞご理解の程お願いします。

Tape MZ-A

MZ-1200/80K/C/700対応

2月1日 Vol.1発売

¥3,000(ブック型パッケージ包装)

DiskPC88

3月Vol.2 発売予定

¥3,000(5″ディスケット1枚)

●Disk FM、Tape MZ-A、Disk PC88は、鈴木技研のオリジナルソフトです。
●通信販売は、送料、他で200円増しになります。

●Disk Series 販売代理店

アサマ電子㈱☎03(251)4700

㈱コーサカ☎06(763)5801

●お申込・お問合せは

鈴木技研SE部

〒354 埼玉県富士見市三芳郵便局私書箱28号

☎0492(52)2513

ツクモ 全店 棚卸し決算大バーゲン

1月10日〜30日まで

ツクモトレード センター開設!!

貴方がパソコンを売りたい・買いたいと思った時、ツクモトレードセンターにお問い合わせ下さい。当センターで動作状態が完動か整備かをチェックし、商品を充分査定の上お客様の希望価格を設定してツクモトレードセンター中古品リストに登録致します。

1 受付 お電話かニューセンター店にてお申し込み下さい。
［売りたい人］ 直接品物をお持ち頂いても結構です。その場で打ち合わせをし、トレードセンターお客様カードに内容等記入致します。お電話でお申し込みになった場合もお客様カードを記入し、送付致しますので条件確認の上、サイン及び捺印して頂き申し込み金￥2,000(トレード成立後返金致します)を同封の上、返送して下さるようお願い致します。
［買いたい人］ 委託者同様ニューセンター店又は電話で受付致します。お客様カードを承認の上、記入・捺印して頂き、送付してもらいます。

2 取引方法 確認されたお客様カードが着き次第御連絡致します。その時に商品を当センターに送付して下さい。ツクモトレードセンター技術スタッフが動作状態等査定し、再度御連絡して希望価格を設定致します。(この場合、販売価格の変更も有りますので御了承下さい。)
お預かりした商品は店頭又は電話にて購入希望者を受け付け致します。希望先着順にお電話で商品内容を出来るだけ細かく説明致します。

■査定基準
1 メーカー及び機種名、S/No.セットバージョン 2 使用年月 3 動作状態(完動か整備が必要か) 4 マニュアル、ケーブル等付属品の有無 5 オリジナルダンボール及びパッキンの有無

■トレード成立後代金は……
委託者の条件で手数料(含む振り込み手数料)を差し引いた金額をすみやかに振り込みます。

●梱包状態不良による輸送中の事故は当社では責任を負いかねます。

12/20〜1/20迄コレクトコールによる受付も致します。

ツクモトレードセンター☎03-251-9977

だからツクモで買う…
ツクモウルトラシステム

大特価クレジット	頭金ナシ
魅力の価格がクレジットで	月々￥3,000クレジット
日曜配達	**支払いは約2ヶ月後**
(札幌店除く) より楽しく日曜日	スキップクレジットで
低金利クレジット	**全商品保証付**
便利なクレジット60回までOK!	製品はすべてメーカー保証付
高額下取制度	**電話クレジット**
ツクモ全店高額下取りOK!	お申し込みは電話一本でOK!

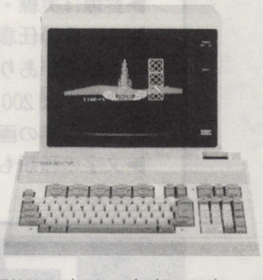
★今、お申し込みになればお支払いはあなたのご希望に
(但し6ヶ月以内)

★ツクモらくらくクレジットお問い合わせは☎251-9911まで

頭金ゼロ・毎月のお支払いもゼロ
お支払いはボーナス2回払い!
ツクモウルトラクレジット

話題の新製品MSX・16ビットが続々入荷!!

富士通FM-7 組み合わせ 本体標準価格￥126,000



★FM-7組み合わせ

Aセット FM-7(本体￥126,000)+TVアダプター(￥12,600)+データレコーダ￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥165,800➡特価**￥125,800**

Bセット FM-7(本体￥126,000)+TH14NM2G(14"1000文字 カラーCRT￥69,800)+ケーブル(￥1,500)+データレコーダー￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥220,400➡特価**￥148,800**

Cセット FM-7(本体￥126,000)+NH14DD(14"2000文字カラーCRT￥69,800)+ケーブル(￥1,500)+データレコーダー￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥238,000➡特価**￥167,800**

Dセット FM-7(本体￥126,000)+VISION II FM(12"2000文字カラーCRT￥79,800)+ケーブル(￥1,500)+データレコーダー￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥248,600➡特価**￥177,800**

Eセット FM-7(本体￥126,000)+NH-12D9(12"2000文字カラーCRT￥99,800)+ケーブル(￥1,500)+データレコーダー￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥268,600➡特価**￥179,800**

Fセット FM-7(本体￥126,000)+TH12M2(MB27303相当12"2000文字￥99,800)+ケーブル(￥1,500)+データレコーダー￥12,800+オリジナルソフト5本(￥15,000)
=合計定価￥268,600➡特価**￥192,800**

★FM-7+プリンタ組み合わせ

Gセット FM-7(本体￥126,000)+TH12M2(MB27303相当12"2000文字￥99,800)+カラーCRTケーブル(￥1,500)+MB27406(プリンター￥89,000)+プリンターケーブル(￥6,000)+オリジナルソフト5本(￥15,000)+データレコーダ￥12,800
=合計定価￥350,100➡特価**￥263,800**〈プリンター用紙サービス中〉

★FM-7+DISK組み合わせ

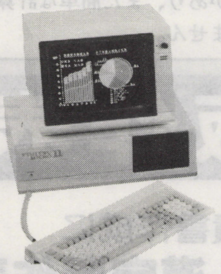
Hセット FM-7(本体￥126,000)+NH12D9(12"2000文字￥99,800)+KD290/FM(￥148,000)・K-305FM(￥128,000)・NF-210FM(￥138,000)・TF10F(￥125,800)のうちいずれか1点+ディスクインターフェースケーブル(￥15,000)+システムディスク(Disk BASIC￥7,000)+オリジナルDISKソフト6本入り1枚(￥9,800)+TS2D10枚(生ディスケット￥15,000)
=合計定価￥395,400〜￥423,600➡特価**￥314,800**

★FM-7+ワープロ組み合わせ

Iセット 【TAPE版ワープロ】FM-7(本体￥126,000)+NH14DD(14"2000文字カラーCRT￥69,800)+カラーCRTケーブル(￥1,500)+データレコーダー￥12,800+MB27406(プリンター￥89,000)+プリンターケーブル(￥6,000)+F漢(TAPEワープロソフト￥7,500)+オリジナルソフト5本(￥15,000) =合計定価￥327,600➡特価**￥244,800**〈プリンター用紙サービス中〉

Jセット 【DISK版ワープロ】FM-7(本体￥126,000)+RT1420Ⓕ(14"2000文字CRT￥98,000)+カラーCRTケーブル(￥1,500)+KD290/FM(￥148,000)・K-305FM(￥128,000)・NF-210FM(￥138,000)・TF10F(￥125,800)のうちいずれか1点+ディスクI/Fケーブル+システムディスク(Disk ASIC￥7,000)+MB27406(プリンター￥89,000)+プリンターケーブル(￥6,000)+FMJWP/7(SBCワープロ￥20,000)+オリジナルDISKソフト6本入1枚(￥9,800)+TS2D10枚(生ディスケット￥15,000)=合計定価￥510,100〜￥538,300➡特価**￥408,500**〈プリンター用紙サービス中〉

富士通FM-11組み合わせ



FM-11AD+漢字ROMカード+NH14DE(640×400ドット400文字)+ケーブル+ディスケット10枚+ディスク版ゲームソフト(5種類入)
=合計定価￥525,600➡特価**￥419,800**

★FM-11ADをEXタイプに変更の場合
=合計定価￥585,600➡特価**￥469,800**

★DISKをデュアルドライブにした場合には￥48,000を加算して下さい。

●FM-11には簡易言語「FMCALC」が付属しており、本格的ビジネス分野に利用できます。
●漢字ROMカードには日本語ワード・プロセッサーが付属しています。

各定価￥2,500をプレゼント!!

★FM-7お買上げの方
RAM DISK プログラム(裏RAMをディスクとして使うソフト)

★FM-7プリンタお買上げの方
プリンタバッファプログラム(裏RAMをプリンタバッファとして使うソフト)

★FM用ディスク又はFM-11お買い上げの方
インターリーブフォーマットコンバータ(ディスクのアクセスが2〜3倍速くなるフォーマットプログラム)

通信販売のお問い合わせは
03(251)9911へ
ご注文の際は、ご注文になりたい商品名をご記入の上、現金書留にてお願い致します。20万円以上は銀行振込みも取り扱っております。お問い合わせ下さい。
富士銀行 神田支店
普通口座 No.894047

即決 ツクモ全国クレジットOK!!
●現金特別価格で各種クレジットが利用できます。残金のみに金利がかかります。
●60回払いまで出来ます。但、1回の支払い額は3,000円以上。
●その場で、お持ち帰りできるクレジットもあります。
●印鑑、身分証明書(免許証など)が必要です。尚、学生の方及び、未成年の方は、ご両親の保証も必要です。
●取扱いクレジットカード:日本信販、JCB、DC、UC、VISA、MC、CF
尚、クレジットカードご使用の場合は、手数料がかかります。

■ご注文商品の中、未着の品物がございました時は、通販受付電話で納品書No.をお知らせ下さい。尚、商品の詳しい取り扱い等は、各販売店までお問い合わせ下さい。

■通信販売は 〒101 東京都千代田区神田郵便局私書箱135 九十九電機㈱ Oh!/FM2係

九十九(ツクモ)電機 株式会社

ニューセンター店	〒101	東京都千代田区外神田1-16-10	☎03(251)0986〜7
名古屋店 3F	〒460	名古屋市中区大須3-30-86 ラジオセンターアメ横ビル内	☎052(263)1681
5号店	〒101	東京都千代田区外神田3-1-14	☎03(251)0531〜2
7号店	〒101	東京都千代田区外神田1-9-9	
札幌店	〒060	札幌市中央区南二条西3-15-1 さっしんビルB1	☎011(241)2299
ビデオセンター店	〒060	札幌市中央区南二条西4 乙井ビルB1・1F	☎011(241)3999

■定休日 東京各店は毎週木曜日と第3水曜日、名古屋店は毎週月曜日

RP-80・FP-80対応

シリアルドットプリンタ用

PC-8801 日本語ワードプロセッサ

PC-SWP ¥39,800

▶ 20,000語辞書が標準装備され、なおかつユーザーが登録することが出来ます。
▶漢字への変換方式はローマ字・カナ・16進の3種類の入力が出来ます。
▶罫線は実線・細線・点線・アダーラインの4種類があり、作表が容易に出来ます。
▶ 240件の任意の文字または記号などをユーザーが作成することが出来る外字機能があります。
▶ＣＲＴは200ライン及び400ラインどちらでも使用可能です。
▶横方向への画面スクロールにより最大53桁までの文書の作成が出来ます。
▶文字の種類も倍幅・全角・½文字の3種類あり、また縦書き出力も出来ます。

ＰＣ－ＳＷＰ（スーパーワープロ）種別
・ＰＣ－8822・ＮＫ3618－22用
・ＦＰ－80・ＲＰ－80用
注ＰＣ－8801用ＭＰシリーズでは使用できません。

FM-7決定版 ワープロ

文節単位での変換処理が可能

FM-7専用 日本語ワードプロセッサ

FM-JWP/7 ¥20,000

▶漢字への変換方式にはローマ字・カナ・16進の3種類の入力ができ、また送り仮名付き入出力が可能、いわゆる文節単位での処理が出来ます。
▶ 20,000語辞書が標準装備されているほか、127パターンの外字登録機能があります。
▶ 1ドライブ及び2ドライブのどちらをお持ちの方でも、機種選択処理によりお使いになれます。
▶罫線は実線・細線・点線・アンダーラインの4種類があり、作表が容易に出来ます。
▶文字の種類には倍角・全角・½文字の3種類があり、また縦書き出力も出来ます。
▶移動・複写などを含む豊富な編集機能があり、また簡単な計算処理も可能です。
※本プログラムはＦＭ－８では使用出来ません。

漢字版作表簡易言語

集計表・一覧表・見積書作成に

FM-7専用 漢字版作表簡易言語

FM-PLAIN/7 ¥20,000

▶日本語ワードプロセッサＦＭ－ＪＷＰ／７と同様、カナ漢字変換方式により漢字の入力がとても簡単です。
▶操作性を重視して、処理メニュー方式、ガイダンス表示、操作手順方式などを採用、コンピュータに不慣れな方でも簡単に操作でき集計表、一覧表、見積書などが誰でも簡単に作成できます。
▶罫線の入力も自動罫線入力機能により非常にスピーディーです。
▶四則演算、分類、検索、複写など豊富な機能を備えており、幅広い用途が可能です。
▶ひとつの表について、最大、横26列・縦60行の項目を持ち、用途にあわせて任意の大きさの表が作成できます。
▶表の印刷は範囲指定ができ、必要な部分だけ印刷することができます。

ＳＢＣお手伝いさんシリーズ

お寺さんのお手伝い

（ＦＭ－７・８用檀家管理システム）
基本部 ￥150,000　オプション部 ￥50,000

- 1000軒の住所・氏名等を出力でき、回忌の案内及び明細、宛名印刷ができます。
- お布施、寄付金、年会費等の実績管理が出来ます。
- 過去台帳の登録、檀信徒の登録、弊社にて承ります。（有償）

酒屋さんのお手伝い

（PC 8801・PC 9801・酒類小売業システム）
￥250,000

- 酒類小売販売店にとって最も繁雑な業務の税務署への報告書作成及び指定の酒類受払帳の作成等を自動的に行います。
- 出力帳表としては、仕入先別仕入月報・容量別在庫月報・品目別在庫月報・酒類別在庫月報・酒類受払帳等があります。

見積書作成のお手伝い

（FM-７・８・PC 8801・PC 9801 塗装面積計算システム）
￥250,000

- 計算式はどんな式でも最大32式まで自由に登録、修正する事が出来ます。
- 変数項目は最大32項目まで使用出来ます。
- 集計機能があり、各識別記号により、小・中・大計をとる事が出来ます。
- 計算書のタイトル名を付けられ、全て会話形式で作業を進められますので操作が簡単です。
- 本システムは橋梁設計で塗装面積の見積用として開発されたものですが、あらゆる見積作業に対応出来るよう、システムの変更等承っております。御気軽に御相談下さい。

財務会計システム

FM-ZAIMU

（FM７・8・11用DISK版 財務管理システム）三和ビジネス㈱製
・ビジネスプリンター・シリアルドットプリンター
基本部 ￥150,000
オプション部 ￥50,000

- ▶１枚のミニフロッピーディスクに最大10,000件もの仕訳が入力保存できます。さらにミニフロッピーディスクを複数にすることによって、最大14,000件の仕訳伝票処理に対応できます。
- ▶仕訳伝票入力時の摘要入力、勘定科目名の修正等、漢字の入力はカナ漢字変換方式が使用でき、漢字の取り扱いはとても簡単です。
- ▶カナ漢字変換による摘要入力の他に固定摘要のコード入力や更新後の仕訳伝票の修正など随所に豊富な機能を取り入れ、高い操作性を実現しました。

PCA高級財務会計システム

（MZシリーズ・FM-7・8・11 PC 9800）ピーシーエー㈱製
￥98,000 ￥148,000 ￥198,000

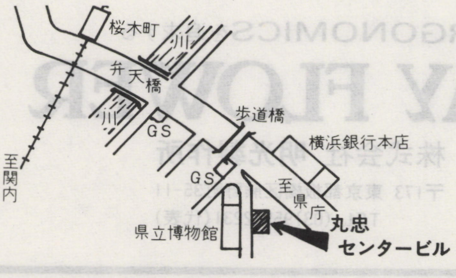
- ▶あなたの会社に合わせた科目登録や補助簿登録ができ、合計金額1,000億円未満の会計処理ができます。
- ▶主要帳票はオフコン並の専用用紙が用意されておりますので、銀行や税務署等にそのまま提出できます。
- ▶画面は漢字の大文字で表示され、非常に見やすくなっています。
- ▶日常良く使われる主要48科目は、ワンタッチ入力方式になっています。
- ▶入出金データーの入力は自動仕訳方式にする等、使いやすく、正確で、しかも主要部分は機械語処理により、トータル・パフォーマンスを追求しています。
- ▶処理範囲によりA．B．Cと３段階のシステムが用意され、必要なレベルを選択できます。
- ▶会計専門家数名のノウハウを結集したシステムです。

PCA給与計算システム

（PC9800）給与計算 ￥59,000　賞与計算 ￥39,000　年末調整 ￥39,000

- ▶月次の給与計算から賞与、年調までの一連システムです。
- ▶月々の変動データを入力するだけで各種帳票が自動的に集計プリントできます。
- ▶各データやマスター類のメンテナンスが容易です。
- ▶オフコン並みの専用用紙も使えます。
- ▶給与計算、年末調整はそれぞれ単独でも使用できます。

パソコン・
プログラマー
募集中!!

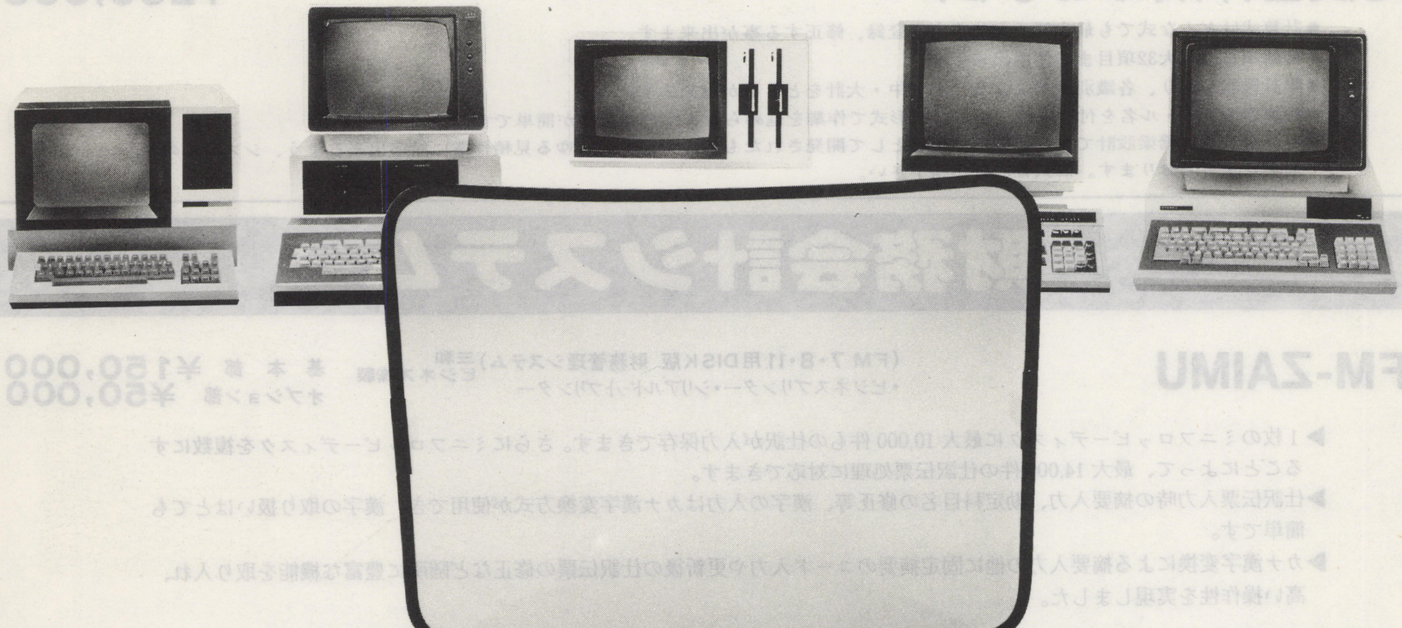


株式会社エス・ビー・シー
〒231
横浜市中区本町５丁目４９番地
丸忠センタービル３Ｆ
TEL 045－201－8480

眼を大切に、だから・・

E·Pスクリーン

Eye Protection Screen



CRTディスプレイ視作業は　なぜ眼が疲れるのか?

(1)私達は、家庭用TVを、慢然と見ていますが、CRTディスプレイ作業では、判読し難い文字を**注視**しなければなりません。それに加えて、作業時の精神的緊張も、疲労の原因と考えられます。

(2)CRT(ブラウン管)画面には、特有のチラツキやマタタキがあり、これらは目を疲れさせるばかりか、中枢神経へ影響を及ぼすともいわれています。

(3)CRT画面の文字は、明るく発光している点の集まりです。至近距離(30〜70cm)で長時間見つめることは、目に負担をかけることになります、。

(4)CRT画面に、外光や天井照明が写り込むと、反射が眩しく、煩わしいものになります。

E.Pスクリーンは、アーゴノミクス(ergonomics)※の見地から研究開発されたものです。従来の透明着色フィルターと違い、当社独自開発のD-M方式を採ることにより、チラツキやマタタキを効果的に押えることができます。スクリーン表面は、外光の反射を緩和するノングレア処理が施されています。又、画面の地と発光体のコントラストを適正にするので、文字をより見やすくします。特にカラーCRTでは、本来のカラーをより鮮やかに際立たせることが出来ます。EPスクリーンは、産業医科大学の実験により、疲労緩和に効果のあることが実証されております。職場の環境整備の一助にEPスクリーンをお役立て下さい。

	CRTの型名	価格		CRTの型名	価格
IBM	5251, 5255	14,500	SORD	DUS-12G	13,800
	3278	14,500		#223	13,800
	6580	14,000		#243	13,800
富士通	6262R	14,500		#243C	15,000
	6262K	16,000	XEROX	14C	16,000
	6652A	14,500	東芝	JW-10	13,800
	9526R	14,500		JW-7	14,500
NEC	PC-8048K	15,000	日立	560-20	14,500
	PC-8049K	15,000		560-20-C	16,000
	PC-8050K	13,800		2171	16,000
	PC-8053	16,000	DEC	VT-100	13,800
	PC-8853	16,000	WANG	DIS-140	13,800
SORD	DUH-14C	16,000	テキサスインスツルメント	DS990-Model-S	13,800

※Ergonomics(人間尊重の立場から、人間の安全と衛生を確保し、人間と機械との調和をめざす科学。)

ERGONOMICSを追求する

MAY FLOWER

株式会社 明光製作所

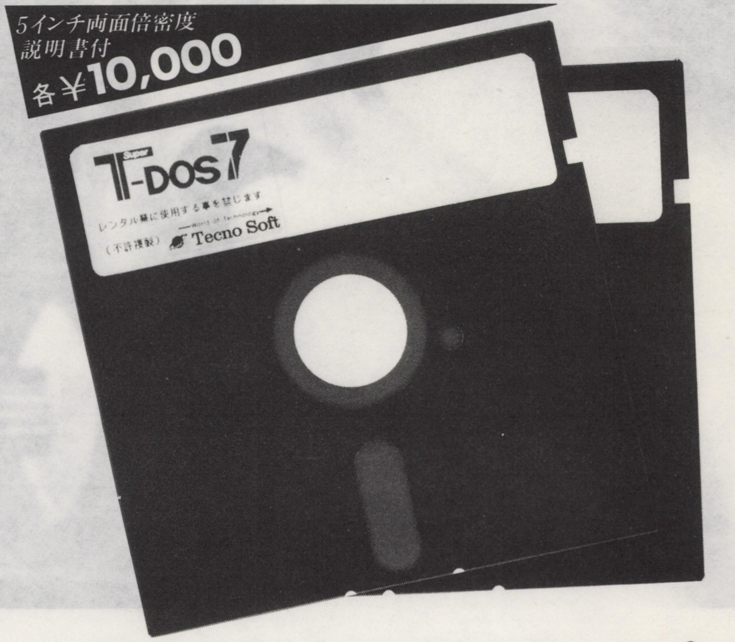
〒173 東京都板橋区熊野町35-11

TEL.(03)955-2231(代表)

Super T-DOS7 (FM-7用) Super T-DOS8 (FM-8用)

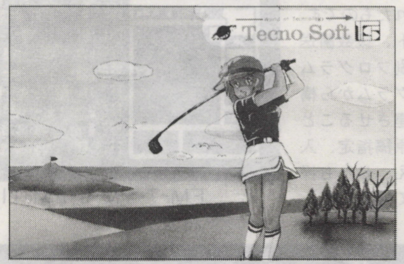
ディスクでROM-BASICと同じフリー・エリアが使えます。

FM-7用：30530Bytes　FM-8用：30358Bytes



①ROM-BASICと同じフリーエリア(30530バイト：FM-7, 30358バイト：FM-8)を保ちつつ，DISKに対して、LOAD/SAVEが可能となりLOADエラーから開放されます。
②ROM-BASICの時の命令は全て使用できます。
③１枚のディスケットには，最高で152のプログラムが収納可。
④PF-KEYの設定をBOOT時に行なえます。
⑤FILESの表示では，縦横に表示されますので見やすい。
⑥命令に変数が使えます。
(例)A$＝"0：TEST":RUN""＋A$
⑦BASICのTEXTのSAVEやASCIIでのSAVE, MERGE, LOAD又マシン語のLOAD，SAVEその他多数の命令が使えます。
⑧データをカセットにSAVEしていたものが，そのまま又はごく少しの修正でDISKに対して，SAVE，LOADできます。
(例)顧客管理，在庫管理，スケジュール管理，成績処理等。
⑨テープ版で番地不明のマシン語がディスクへ対してSAVE，LOADできます。

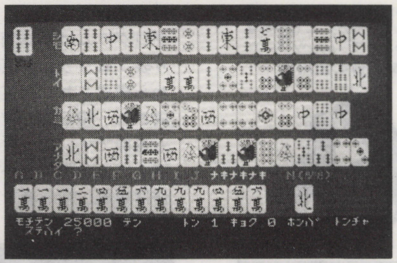
ゴルフ・アイランド



ショット時に必要な情報は全て表示されます。表現もリアルです。(木に当ればの葉が落ちる。水に入れば水しぶきや水音がする等)。また高速です。樹木、池、バンカーの位置、風向、風速はランダムに表示されますから同一ホールでも違ったゲームが楽しめます。

発売機種
FM-7、X1、PC-8801、PC-8001MKII、PC-6001MKII、MZ-2000(Hu-BASIC、G-RAM1、2、3)…………各￥4,000
〔拡張コース〕1、2、3(各￥2,000)も好評発売中。

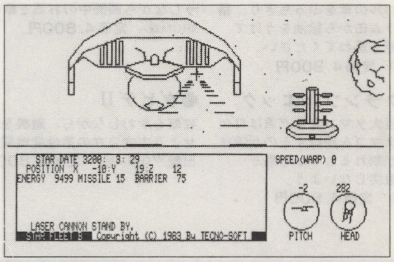
四人麻雀



究極のグラフィック化(PCG)マージャン完成／トンナン回しの半チャンジメで、リーチ、ポン、チー、裏ドラー発あり。コンピューター側の思考はマシン語を採用。(BASIC＋マシン語)。

発売機種
FM-7、PC-8801、PC-8001MKII、X1、PC-6001MKII、MZ-2200(dB-BASIC用)
…………各￥4,000
MZ-700…………各￥3,500

スターフリート/B



驚異の３次元グラフィックが、リアルなコントロール感覚を君に教える。X-Y平面とZ軸をそれぞれ座標で解析、傾きと方向を即座にメーター標示。そして各種センサー、オートパイロット機構が、君の乗るエンタープライズ号を、縦横無尽に操らせてくれる／宇宙空間を高速で迫りくるエイリゴン艦を、君のテクニックで撃破せよ／体が宙に浮かぶ感覚が、君を新体験に導く。

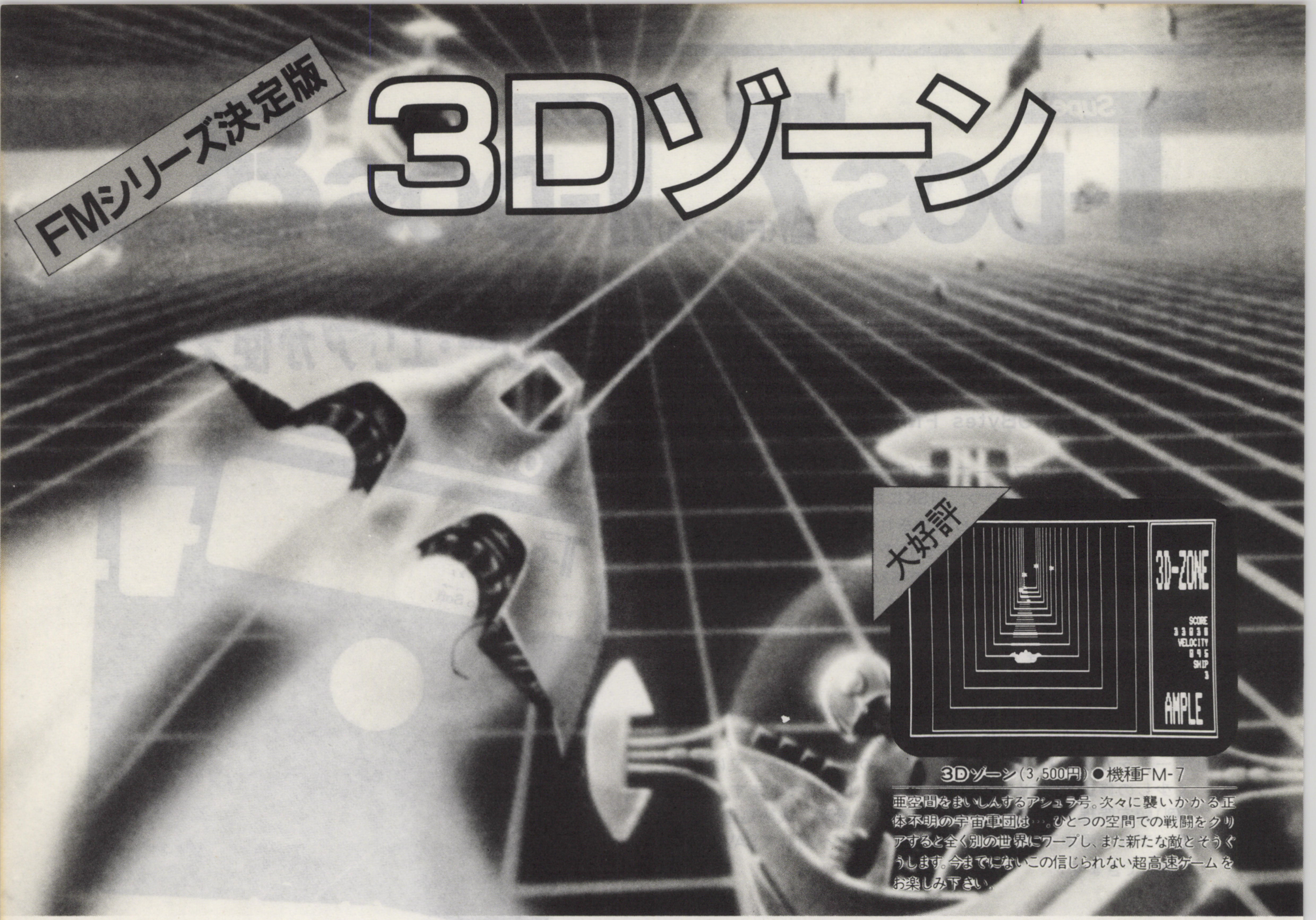
発売機種
FM7/8、PC-8801、PC-8001MKII、PC-6001MKII、X1(G-RAM)、MZ-80B(G-RAM1)、MZ-2000(G-RAM1)。
…………各￥4,500

㈱テクノソフト　AM10:00～PM6:00　日曜定休
〒857 長崎県佐世保市福石町4-14 ☎0956-33-5555
— World of Technology →
Tecno Soft

■皆様からの御意見・御希望をお寄せ下さい。
■お求めは，お近くのマイコン・ショップか，又は直接当社へ，マイコン機種名，品名と電話番号を記入の上現金書留にてお送り下さい。(全種 送料無料)
■Tecno Soft代理店

東京：日本ソフトバンク…………	☎ 03-263-3690
大阪：日本ソフトバンク…………	☎ 06-644-0191
東京：コムパック…………	☎ 03-375-3401
大阪：近畿システィムサービス……	☎ 06-644-6625

FMシリーズ決定版
3Dゾーン



3Dゾーン(3,500円)●機種FM-7

亜空間をまいしんするアシュラ号。次々に襲いかかる正体不明の宇宙軍団は……。ひとつの空間での戦闘をクリアすると全く別の世界にワープし、また新たな敵とそうぐうします。今までにないこの信じられない超高速ゲームをお楽しみ下さい。

MSXカートリッジゲームシリーズ

●CAR RACE
アクセルとハンドルをうまく操ってライバルの車をぶっちぎり。路上のドラム缶から給油をうけて、走行距離を重ねてください。
〈ROM版〉 定価4,800円

●SUPER, DRINKER
お酒を飲むほど元気一杯!!フラフラしながら画面中のお酒を飲め。
〈ROM版〉 定価4,800円

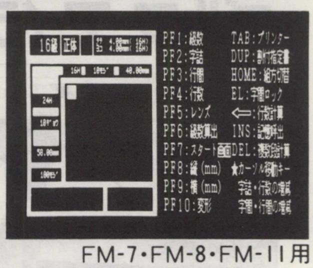
●ブギウギジャングル
友達の家に行こうとするワンちゃん一匹。ジャングルに迷い込んじゃった。こわーい番人をかわして、レッツゴー!!
〈ROM版〉定価4,800円

●スクランブルエッグ
空飛ぶ巨大タマゴ発見!!君はロケットでタマゴを粉砕するのが使命。タマゴが割れると中から鳥が…。地上に激突しないよう。
〈ROM版〉 定価4,800円

●ダビデII
攻撃をかわしながら、敵機を爆破せよ八方向自在の高性能戦闘機で出撃!!〈ROM版〉 定価4,800円

写植レイアウト

このプログラムは、写植文字を頻繁に使用する人達のために、"級""歯"を使わずに、割付指定ができるように考案されたものです。プログラムは、誌面レイアウト、級数歯送り表プログラム、原稿用紙印刷プログラムから構成され、相互間のプログラムから構成され、相互間のデータを関連させることも可能です。レイアウトから原稿指定、入稿といった作業を一貫して行え、しかも操作は簡単、初心者でも安心して扱えます。



FM-7・FM-8・FM-11用

構成

●ソフトウェア
写植レイアウトプログラム
写植プログラム(級数歯送り表プログラム)／誌面レイアウトプログラム(行数・ミリ)／プリンタ出力プログラム(割付指定書・原稿用紙印刷、ミニラフスケッチ)／詳細マニュアル(B5版約200頁)写植テキスト
ソフトウェア価格……………………………………………………定価268,800円

●写植レイアウトシステム(FM-7の構成例)
(ソフトウェア＋ハードウェア)
写植レイアウトプログラム
富士通FM-7本体(MB25010)／カラーCRTディスプレイ(MB27307)／カラーCRT用ケーブル(MB26512)／薄型ミニFDユニット(MB27607)／FM-7用ミニフロッピィインターフェースカード(MB22407)／FM-7用漢字ROMカード(MB22405)／EPSONプリンタFP-80／プリンタ用ケーブル(MB26514)
システム合計価格………………………………………………定価800,000円

いんふぉめーしょん

やあ！元気でパソコンしてますか？ アンプルでは、楽しいゲームをユーザーのみなさんに、今までよりもずっと手軽にもとめていただけるよう"テレフォン・オーダーシステム"というものを始めました。

君は電話を一本するだけ。すばやく君の手許にお届けします。送料は無料。代金は商品が着いてから一週間以内に現金書留か銀行振込で支払えばOK。

また、ハガキでの申込みも受け付けております。もちろん、今までの通り直接、現金書留での注文もだいじょうぶです。

オーダーホットラインは ☎03(466)3170 販売係

＊18才未満の方はご両親の承諾をもらってからお申し込み下さい。なお、商品の性質上、返品または、お取替えはできませんのでご了承下さい。

アンプルソフトウェア株式会社
〒151 東京都渋谷区元代々木町14-3 FAX03-466-3187 ☎466-3170(代)

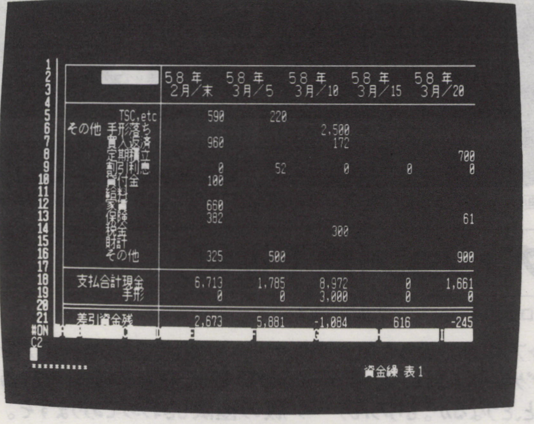
技術・開発スタッフ
イラストレーター
募集中!!

当社のソフトウェアは、全国のパソコンショップでお買い求めいただけます。また現金書留でご購入の方や資料ご希望の方は、当社まで直接お申し込み下さい。

資料請求券 oh!FM 2

作表簡易言語シリーズ

FMCALC



待ってました!!

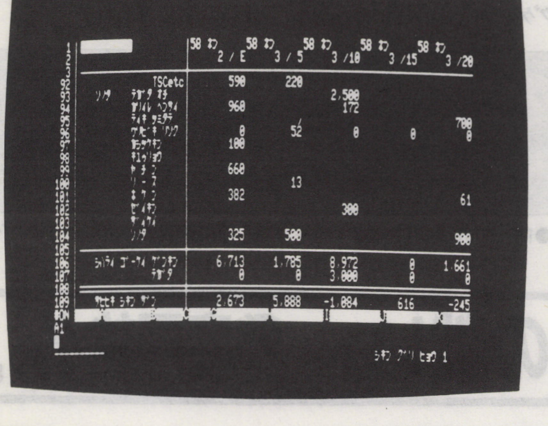
漢字FMCALCはFM-11添付ソフトの「FMCALC」のパワーアップ版です。もちろんデータも共有できます。

新登場

FM-11用

漢字FMCALC

ディスクバージョン……¥58,000



簡潔な命令で必要十分な処理を!!

FMCALCは集計・ソート作業に威力を発揮するFMシリーズ用の本格的ビジネスソフトです。

FMCALC FM-7用 FM-8用

ディスクバージョン……¥48,000

カセットバージョン……¥15,000

FMCALCは管理職の方のためのパーソナルツールです

今日、OA(オフィスオートメーション)の普及により大量のデータ処理が可能となりました。しかし一般管理職や中堅技術者の方々が、自分の業務のためにコンピュータを個人的に使うには難しい問題があります。FMCALCはこの問題を一挙に解決。FORTRAN、COBOL、BASICなど電算用の言語を学ばずに、すぐさまデータを入力し、機械に計算させ、結果を画面から読みとり、またプリントアウトすることができます。

FMCALCは作表専用ツールです

管理業務には沢山の数値データを取り扱うことが避けられません。また多くの場合、数値データを表の形にまとめ、縦横の方向に和をとったり、平均したり、構成比を算出したりすることが必要となります。FMCALCは2〜3のキー操作をするだけで、白紙の表を画面に表示します。あなたは画面を見ながら必要な数値や文字や式を必要な場所に次々とキーボードから入れていくだけで、希望の表を作ることができます。

現状の把握や将来の予測に

FMCALCで作成された表は、複雑な経路を辿って入力されたデータではなく、あなたが直接入力したデータに基づいています。たとえば実数値とは違うが、将来の予測に役立ちそうな数値を試みに入力し、その結果、その他の数値がどんな風に動くかを確かめることができます。したがって販売努力の目標設定や、資金繰の予測などに威力を発揮します。

FMCALCは多様な可能性を秘めています

FMCALCは簡便なソフトウェアですが、深みのあるプログラムです。あなたの創意と工夫を十分に受けとめ、縦横にその能力を発揮する可能性が秘められています。プログラミング言語という媒介を経ずに、あなたは直接いろいろな数値をプランニングし、FMCALCに指示することができるのです。

FMCALCワークシート新発売!

- FMCALCをさらに使いやすくするように作られました。
- FMCALCとワークシートによって、貴方のFMシリーズを本格的ビジネス用パソコンとして、その能力を縦横に発揮できます。

FMCALCワークシート1(売掛管理表、買掛管理表)	¥9,800
FMCALCワークシート2(手形管理表)	¥9,800
FMCALCワークシート3(資金繰表)	¥9,800
★ワークシート1,2,3セット価格	¥25,000
★CALC-PACK ワークシート1,2,3+FMCALC	¥68,000

SMart SOFT MART, inc.

TSC社日本総代理店・スーパーソフト社 JAPAN AUTHORIZED DISTRIBUTOR

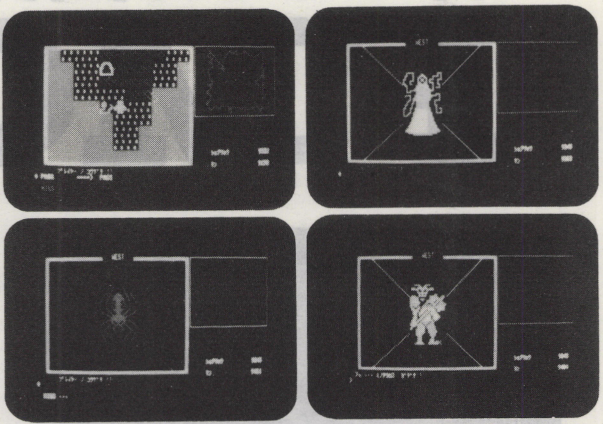
ソフトマート株式会社

〒101 東京都千代田区神田須田町1-18-6 第1谷ビル ☎(03)256-5881 FAX(03)256-6180 GII/GIII

通信販売のお知らせ

α言語、FMCALCをはじめ取扱商品の通信販売をいたします。品名、住所、電話、氏名を明記の上、現金書留にてお申込み下さい。

ソフトのグレードが、マニ



画面写真PC-8001用(PC-8801、FM7のグラフィックスに絶対の自身あり)

ダンジョン(地下迷宮)

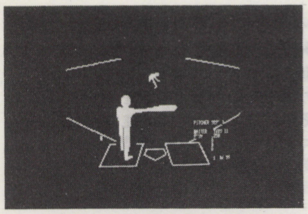
プロデューサー　シブサワ コウ

今アメリカで最も人気があり、ゲームの主流であるロールプレイングゲーム。これを思考ゲームの殿堂光栄マイコンシステムが創ると、どうなるか。もちろんテープ版の極限もきわめてありますぞ。さあダンジョンへ旅立つ貴方、光栄のファンタジーロールプレイングゲームでこのだいごみを十二分にお楽しみあれ！

ダンジョンは、日本のコンピュータゲーマー達にロールプレイングゲームの存在を身近に感じさせてくれる格好のソフトだ。ゲーム進行も比較的シンプルだし、ストーリーも日本人にとって理解しやすい。ロールプレイングゲーム入門に最適。(ログインより)

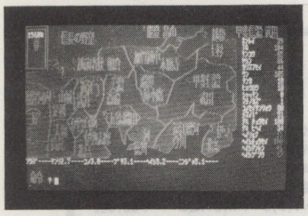
●PC-8001、8801、FM7　テープ版5,200円(ユーザーステープ付)

すごい！
ヘクス画面がスクロール「アルデンヌの戦い」　**発売開始!!**
PC-8801　テープ¥3,800



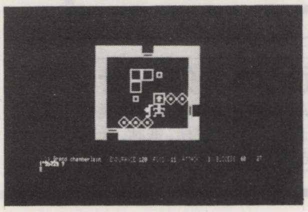
ペナントレース

●テープ4,500円●5'ディスク6,800円
●機種PC-8001/8801/FM7
秋葉原サトー無線様、連続8週1位獲得
その他多数ショップ様にてトップランキング続出！



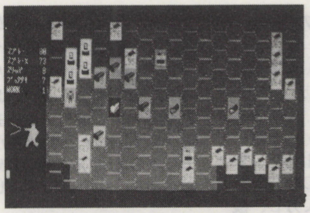
信長の野望

●テープ4,500円●5'ディスク6,800円
(PC-9801用8'ディスク7,800円)●機種PC-8001/8801/9801/9801F/FM7/8/11
"もう目が血走っちゃいそう！ 信長につき合って二晩徹夜してしまったと言ったら読者は笑うだろうか。しかしこれホントの話"。「ポプコム」より



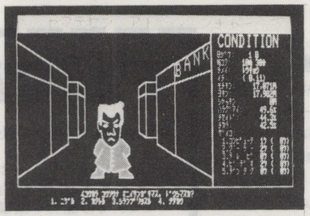
クフ王の秘密

●テープ4,500円●機種PC-8001/8801/FM7
国内第一号のロールプレイングゲーム
"パソコンゲーム界の天才シブサワコウ作。



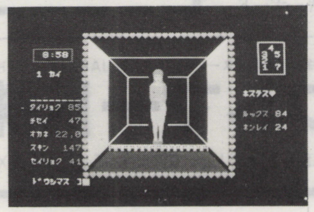
ホイホイ

●テープ3,500円 PC-9801用ディスクパック2本組7,800円
●機種PC-8801/FM7
"爆笑度100%のパロディ精神に脱帽！ 手法はあくまで正統的なウォーゲームを踏え、コミカルゲームに仕立てあげたこの手腕。さすが光栄さんエライ。文句なしのお買得の秀作といえよう!!" 「パソコンゲームランキングブック」より



アドベンチャーパック

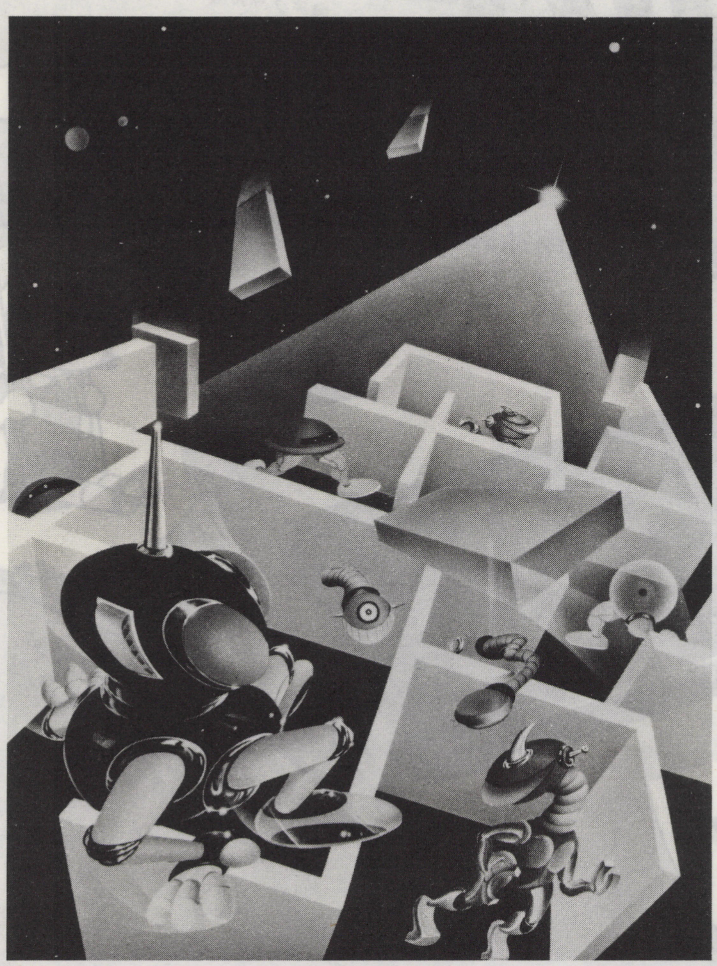
●テープ3,800円●機種PC-8001/8801/FM7/8
"山手線アドベンチャー"、"コンピュータジャック"。二本のグラフィックアドベンチャーゲームが入った超お買得品。

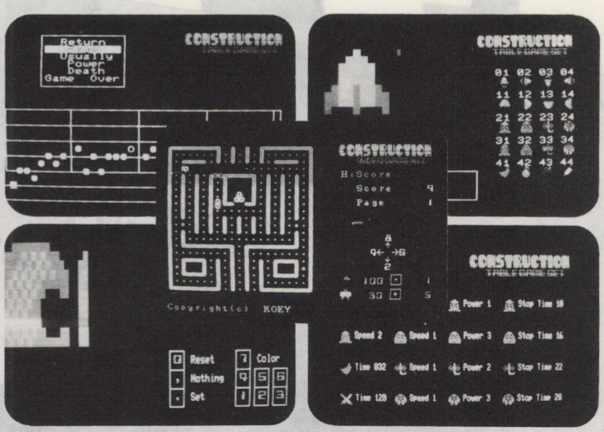


団地妻の誘惑

●テープ4,800円 ●5'ディスク6,800円 (PC-9801用8'ディスク7,800円)●機種PC-8801/9801/9801F/FM7/8(ディスク版FM7/8)
"ひょっとしたらこのゲームはノルマに追われる我々現代人の生活への作者の強烈な文明批判ではないかという気さえしてくる。ゲームとしてはグラフィックや様々なアイディアで充分楽しめる上出来のソフト。"「カーソル」より

アのニーズを越えた。





画面はPC-8001用(PC-8801、FM7のグラフィックは最高)

CONSTRUCTION

プロデューサー 藤田敏夫

まずは、すばらしいアクションゲーム2本で
存分にお楽しみ下さい。
「コンストラクション」とは……
★ゲームの主人公たるキャラクターのデザインを簡単に作れる。
★アクションの設定から迷路の構成も自由自在。
★バックミュージックの作曲、編曲もご自由に。
★その他ゲームに必要な全ての部分を簡単なキー操作一つでクリエイトできる画期的なソフトです。

●PC-8801 テープ版2本組 **5,800円**

CONSTRUCTIONで貴方が作ったゲーム募集!
コンストラクション
1席10万円1名・2席5万円1名・3席3万円1名・佳作10名
くわしくはマニアルをご覧下さい。〆切2月末日消印有効:発表5月号

"話題の名作"
国内発のパロディーシミュレーションゲーム **「ホイホイ」** **FM-7・X1遂に完成!**
PC-8801 テープ¥3,500

おみせできなくて
ゴメンナサイ!

ナイトライフ
●テープ4,500円●5'ディスク7,800円
(PC-9801用8'ディスク8,800円)●機種
PC-6001MkII/8001/8001MkII/8801/
9801/9801F/FM7/8/11/MZ2200/5500
テレビ、新聞・雑誌にと、これ程話題に上ったソフトは、かつてなかった。世界に類を見ないアダルトの世界。

ビジネスマン必携!! ビジネスシミュレーションゲーム
「トップマネジメント」
●5インチ、8インチディスク版●カナ版16,800円●漢字版19,800円
(PC-9801F用のみ)

製造、販売、財務等、会社経営の全てをシミュレートした本格的マネジメントゲームです。経営者として日々決断すべき事柄の大部分を網羅してありますので、自己啓発用、企業の社員教育用に是非御活用下さい。

〈ディスク版〉
PC-8001、PC-8001MkII、PC-8801、PC-8801MkII、PC-9801、PC-9801F
(PC-9801F漢字版、FM7、FM8、FM11、開発中)

日本語ワードプロセッサー PC-8001
「い・ろ・は」
PC-8001で漢字!
★ソフトバンク日本語ワープロ出荷ベース第5位
PC-8801・PC-8001 MKII(N-BASIC可) **29,800円**

●マシン語による高速処理●漢字ROM不要●漢字プリンタ不要●本体改造不要●熟語登録1万語●拡大文字・アンダーライン・縦書き機能その他豊富
▶使用可能プリンター/PC-8023C/PC-8821/PC-8822/MP-80II/MP-80III/RP-80/FP-80/FP-80K/GP-500E/GP-250F

汎用漢字顧客管理システム ★日経パソコンベストヒット商品ランキング第5位
「商売繁盛」
V2・0 FM-7/8 **29,800円**

●1ディスケットあたり600件の顧客情報●仮名漢字変換方式により簡単に漢字登録●業種に合せて、ユーザー独自の項目設定可能●メニュー表示により、選択できるので操作簡単、その他豊富な機能盛りだくさん
▶使用可能プリンター/MB-27403/MB-27404/MB-27405/MP-80II/MP-80III/RP-80/FP-80/FP-80K/GP-250F/GP-550E/MP-80K/PC-8822

光栄マイコンシステム★

本社:〒326 足利市通1丁目2677 ☎0284-41-5911㈹
支社:〒233 横浜市港北区日吉本町1876 ☎044-61-6861㈹

■お求めは全国のマイコンショップ又は本社宛現金書留にてお送り下さい。(送料サービス)

カタログ請求券
FM 2月号

あなたのパソコンを うつくしくよそおう

キーボードカバー

- ●美しくキーボードをカバーします
- ●デモ中に他の人の手が触れるのをガードします
- ●アクリル製

富士通パソコン用

●カラー：ブラウンスモーク

KEYBOARD COVER

FM-7
FM-8
FM-11
FM-X MSX用新発売

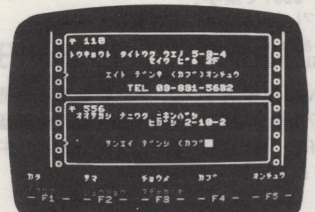
イニシャルプレゼント

キーボードカバーお買いあげの方にもれなくイニシャルシートをプレゼントいたします

宛名書き専用ソフト（住所録・電話帳）

ゆうびん屋さん

12月20日 新発売



手書きの苦労がふっとんだ!!

「ゆうびん屋さん」は機能を限定して、めんどうな操作や設定を除いた結果、はじめての方でも簡単に使っていただけるようになりました。

FUJITSU FM-7 型番 SF-501 **¥3,800**

マイコン用

宛名 タックシール

（型番 TC-10）

- ●4インチ幅トラクタフィールド用紙
- ●1ケース・1ロール500枚入×2コ入
- ●シールサイズ（タテ）38mm×（ヨコ）86mm

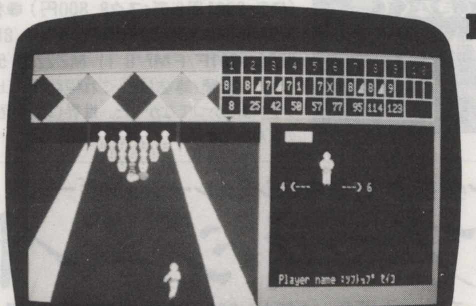
¥3,200

ユニークな発想のゲームソフト

スーパーボウル

FUJITSU FM-7

型番 SF-001
¥3,500



マリちゃんはスポーツギャル。今日はソフトップボウルにやってきました。テクニックバツグンのマリちゃんも、はじめてのレーンでちょっと緊張。さあ、ハイスコアは何点だ？

■お求めは……　（商品送料はサービスいたします）

全国パソコンショップまたは直接右記へお申し込みください。

- ● 現金書留でご注文の場合は商品名、住所、氏名、電話番号をはっきり書いてお送りください。
- ● 銀行振込みでご注文の場合は商品名、住所、氏名、電話番号をハガキまたは電話でご連絡ください。

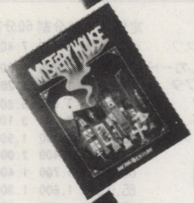
総発売元 **エイト電気株式会社**
〒110 東京都台東区上野5-3-4 ☎03-831-5632（代）
振込銀行：北陸銀行　上野支店（普）No.4032110

西日本発売元 **三栄電子株式会社**
〒556 大阪市浪速区日本橋東2-10-2 ☎06-643-3833（代）
振込銀行：三和銀行　恵美須支店（普）No.189879

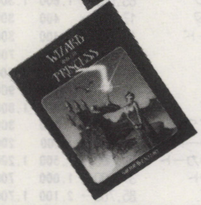
84-1 oh!FM

あのアップルのアドベンチャーゲームが続々FMに!!

■ミッション・アステロイド
小惑星アステロイドが地球に突進してくる。
地球を救うのは、あなたです。
幸運を祈ります。¥7,800



■ミステリー・ハウス
あなたは、ヴィクトリア式の大邸宅の庭にいます。ミステリーと殺人と陰謀が待ちうけています。¥8,800

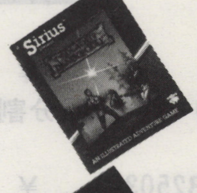


■ウィザード＆プリンセス
山岳地帯のかなた幽閉されているセレニアのプリシラ姫をすくうため、あなたは立ち上る。しかし持っているのは、水筒、ナイフ、パンそして毛布一枚だけです。さて……… ¥12,300

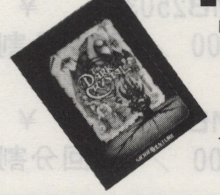
■クランストン・マナー
スランストン卿の強欲のために疲弊し、無人と化したコースゴールドの町で、あなたの目的は屋敷に入り込み隠された16の財宝を捜し出す事です。¥12,800

■ユリシーズ
航海の途中あなたはユリシーズが戦った相手と同じ邪神、怪物と遭遇します。あなたも古代人の白衣をまとい、くさりの鎧を着て、ユリシーズの持てる力をフルに発揮してみて下さい。¥12,800

■トランシルバニア
トランシルバニア地方に住む小悪魔からサブリナ姫を君の手で守らなければ!
¥9,800



■ブラックプールの剣
時は中世。イギリスの片田舎あなたは、落ちぶれた貴族の息子です。自分の手で家運を盛り返せるか!
¥12,300

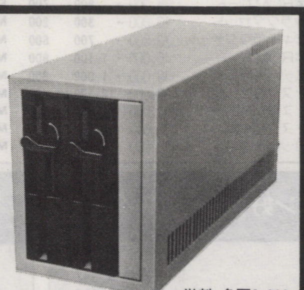


■ダーククリスタル
時は数千年前。平和な世界を守るダーククリスタルがうばわれた。悪魔の手から、ダーククリスタルを取り戻せば、世界に平和がよみ返ります。¥14,800

各種ビジネスソフト・ワープロ・ユーティリティ・ゲーム店内多数展示!!

THIN LINE
ミニフロッピーデュアルドライブ

シンライン／FMシリーズは、富士通純正品と完全にコンパチブルです。さらにFM-7ではI/O・ROMを当社パッチROMと交換することにより、純正品と較べてディスクアクセスを約4倍のスピードで動かすことができます。



送料 各〒2,000

☆FM-8用 ¥143,000
デュアルドライブセット
（フロッピーインターフェイス＋システムディスク含）
●頭金0 月々7,300円×24回
●頭金0 月々5,200円×36回

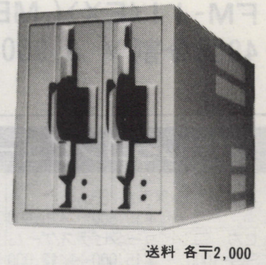
☆FM-7用 ¥141,000
デュアルドライブセット
（フロッピーインターフェイス＋システムディスク含）
●頭金0 月々7,200円×24回
●頭金0 月々5,100円×36回

☆デュアルドライブ ¥99,800
スタンダード
（自作派及増設用に）
●頭金0 月々5,100円×24回
●頭金0 月々3,700円×36回

当社フロッピー取り扱い店
募集中

20% OFF

THIN LINE 8
8インチ
フロッピー
デュアルドライブ

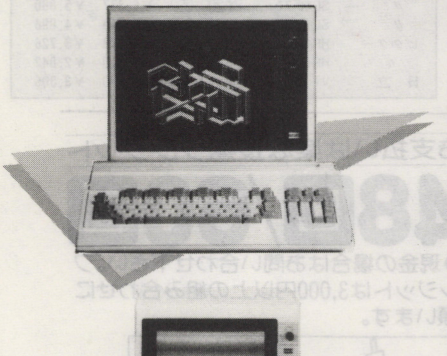


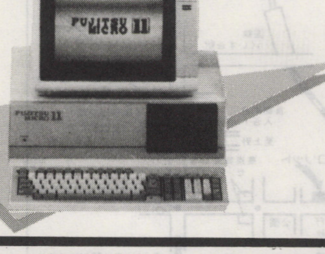
送料 各〒2,000

☆PC-88用セット
（フロッピーコントローラ＋PC-8884含）
定価¥328,000 特価¥262,800
●頭金0 月々13,300円×24回
●頭金0 月々 9,500円×36回

☆PC-98用セット
（PC-9884含）
定価¥286,000 特価¥226,000
●頭金0 月々11,600円×24回
●頭金0 月々 8,300円×36回

ADO マイコンショップ 1F ブックショップ 2F 03-255-9515
〒101 東京都千代田区外神田2-2-1





FM-7 入門用セット

本体	126,000円
グリーンモニター	36,800円
FMデータレコーダ	12,800円
定価	175,600円
特価	135,000円

ボーナス一括払OK!
●頭金0 月々6,900円×24回
●頭金0 月々4,900円×36回

FM-7 ドライブセット

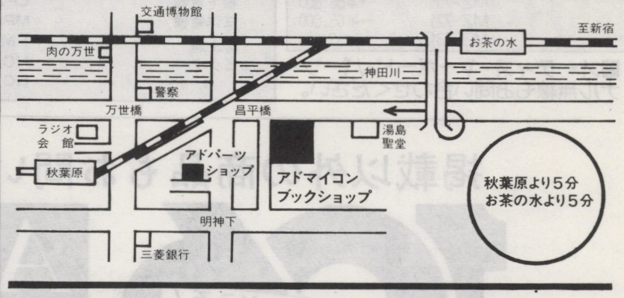
本体	126,000円
THINLINE デュアルドライブセット	141,000円
カラーディスプレイ（ビジョンII）	79,800円
カラーケーブル	1,800円
プリンター（ライトマン）	84,800円
定価	433,400円
特価	376,000円

●頭金0 月々13,600円×36回
●頭金0 月々10,600円×48回

FM-11 EXセット

本体	398,000円
増設ドライブ	65,000円
カラーモニター4050文字	118,000円
カラーケーブル	1,800円
定価	582,800円
特価	490,000円

●頭金0 月々17,700円×36回
●頭金0 月々14,000円×48回



通信販売ご利用のお客様は下記へお問合せ!
もしくは現金書留にてお送り下さい。
〒101 東京都千代田区外神田2-4-6
ビルディングササゲ7F
株式会社 亜土電子工業
通販部 oh!FM係 TEL 03-253-8307
※お願い：住所・氏名・TEL・注文書は明確に、また、お忘れない様!

〔初めて握ぎるハンダゴテ〕用部品セット 好評発売中!!
本誌：第4号61頁〔インターフェースを作ろう〕用

FM PIA SET ¥5,200 Disk 兼用型 ¥5,000
FM LED SET ¥2,400 送料何台でも 〒300

電話で深めようマイコンの世界

106 106 106 106 106 無料 106 106 106 106 106 無料 106 106 106 106 106

アコリットマイコン最新カタログ

富士通FMシリーズ

FM-7(MB25010)　¥ 126,000
48回分割¥ 3,100 ／ 60回分割¥ 2,600

FM-11<ST>(MB25030)　¥ 268,000
48回分割¥ 6,700 ／ 60回分割¥ 5,700

FM-11<AD>(MB25040)　¥ 338,000
48回分割¥ 8,400 ／ 60回分割¥ 7,200

FM-11<EX>(MB25050)　¥ 398,000
48回分割¥ 9,900 ／ 60回分割¥ 8,500

型名	品名	定価 円	48分割	60分割
MB 27312	グリーンディスプレイ	49,800→	1,200	1,000
MB 27302	グリーンディスプレイ	46,800→	1,200	900
MB 27304	グリーンディスプレイ	29,800→	700	600
MB 27331	カラーディスプレイ	188,000→	4,700	4,000
MB 27301	カラーディスプレイ	188,000→	4,700	4,000
MB 27307	カラーディスプレイ	155,000→	3,900	3,300
MB 27303	カラーディスプレイ	99,800→	2,500	2,100
MB 27305	カラーディスプレイ	64,800→	1,600	1,400
MB 22602	家庭用TVアダプタ	13,500→	300	300
MB 27402	ビジネスプリンタ	350,000→	9,000	7,400
MB 26201	フロントインサータ	89,000→	2,200	1,800
MB 27406	シリアルプリンタ	89,000→	2,200	1,800
MB 27404	ドットプリンタ	89,000→	2,200	1,800
MB 27501	データレコーダ	12,800→	300	200
MB 22406	I/Fカード	15,000→	400	300
MB 26001	拡張ユニット	133,000→	3,300	2,800
MB 22204	I/Fモジュール	30,000→	700	600
MB 22203	計測制御I/F・M	61,000→	1,500	1,300
MB 22209	音声入出力M	98,000→	2,400	2,000
MB 22405	漢字ROMカード	35,000→	900	700
MB 22002	キャラクタセット	10,000→	300	200
MB 22003	漢字キャラクタセット	30,000→	700	600
MB 28111	漢字ROMカード	46,000→	1,100	900
SM 07217 F061	システムカセット	39,000→	1,000	800
MB 27651	ディスクユニット	498,000→	12,500	10,600
MB 22416	ディスクI/Fカード	14,000→	400	300
MB 22211	ディスクI/F・M	20,000→	500	400
MB 27603	フロッピディスク	440,000→	11,000	9,400

型名	品名	定価/円	48分割	60分割
MB 27513	フロッピディスク	350,000→	8,700	7,400
MB-22415	Fコントロールカード	30,000→	700	600
MB-22604	フロッピアダプタ	58,000→	1,400	1,200
MB-27061	M/Fディスク	313,000→	7,800	6,600
MB-27605	S/Fディスク	148,000→	3,700	3,100
MB-27606	増設M/Fディスク	90,000→	2,300	1,900
MB-27607	S/Fディスク	98,000→	2,400	2,000
MB-27608	M/Fディスク	69,000→	1,700	1,400
MB-27609	M/Fディスク	65,000→	1,600	1,300
MB-22603	M/Fアダプタ	17,000→	400	300
MB-22407	M/F I/Fカード	14,800→	400	300
MB-28021	Z-80カード	35,000→	900	700
MB-28041	8088カード	80,000→	2,000	1,700
MB-28031	Z-80カード	45,000→	1,100	900
MB-22605	バブルホルダ	89,000→	2,200	1,800
MB-22609	バブルアダプタ	13,000→	300	300
MB-22417	バブルI/Fカード	11,000→	300	200
MB-22608	バブルメモリカード	58,000→	1,500	1,200
FBM-C128GP	バブルカセット	37,000→	1,000	700
MB-22601	バブルホルダ	85,700→	2,100	1,700
FBM-43CP	バブルカセット	35,000→	1,000	700
MB-22401	Z-80カード	11,700→	300	200
MB-28011		50,000→	1,300	1,000
MB-28012	8088カード+5″	99,800→	2,500	2,100
MB-28013	CPM/86®	50,000→	1,300	1,000
MB-28014		99,800→	2,500	2,100
MB-22206	8088拡張RAM・M	49,800→	1,200	1,000
MB-22414A	拡張RAMカード	58,000→	1,500	1,200
MB-22414B	拡張RAMカード	88,000→	2,200	1,800

おもしろコンピュータ

アタリ　システム2800	¥19,800→¥15,800
セガ　コンピュータビデオゲーム	
SG-1000	¥15,000→¥12,000
SC-3000	¥29,800→¥23,800
任天堂　ファミリーコンピュータ	¥14,800→¥11,800

●コレクトコールでのお問い合せは御遠慮ください
●この商品のみは－500円をたして現金書留にてお願い致します

お買得コーナー

■カシオ	EP-1100+ソフト =¥65,000
■NEC	PC-8001(下取品) =¥50,000
■ナショナル	JR-100+RFコンバーター +ソフト3巻=¥30,000
■シャープ	MZ-700シリーズ MZ-711 →¥58,300 MZ-721 →¥65,600 MZ-731 →¥93,400

■オーディオ・キーボード・パーソナル無線もお問い合わせください。

注目のMSX対応各社パソコン好評受付中!

型名	品名	定価/円	15分割
日本楽器製造	本体	49,800→	3,347
YIS-303	本体	64,800→	4,355
YIS-503	FMサウンドシンセサイザユニット	19,800→	1,330
SFG-01	MIDIユニット	12,800→	860
SMD-01	プレイカードセット	12,800→	860
ZPA-01	ミュージックキーボード	17,800→	1,196
YK-01	家庭用カラーTVアダプタ	7,800→	530
RF-01	FMミュージックマクロ	7,800→	530
YRM-11	FM音色プログラム	7,800→	530
YRM-12	FMミュージックコンポーザ	7,800→	530
YRM-15	HX-10D	65,800→	4,422
東芝"パソピアQ"	HX-10S	55,800→	3,750
日立	MB-H1	62,800→	4,220
〃	MB-H1E	54,800→	3,683
三菱	ML-8000	59,800→	4,019
ソニー	"HIT-BIT"HB-55	54,800→	3,683
松下電器	CF2000	54,800→	3,683
三洋電機	MPC-10	74,800→	5,027
富士通	FM-X	49,800→	3,347
ビクター	HC-5	59,800→	4,019
〃	HC-6	64,800→	4,355

Oh-FM 愛読者特別セール

テレビ		標準価格	48回均等	60回均等
ナショナル	TH21-H55GR	¥219,000	¥5,037	¥4,292
〃	TH19-L55GR	¥188,000	¥4,324	¥3,684
〃	TH15-M55GR	¥139,000	¥3,197	¥2,724
〃	TH14-NV1	¥208,000	¥4,605	¥3,924
ソニー	KV-20XR2	¥199,800	¥4,481	¥3,818
〃	KV-18XR2	¥179,800	¥4,032	¥3,436
ビクター	AV-260	¥290,000	¥6,527	¥5,562
〃	AV-200	¥210,000	¥4,709	¥4,013
東芝	21K691	¥215,000	¥4,821	¥4,108
〃	15K31	¥129,800	¥2,910	¥2,480
シャープ	C-1 19型	¥145,000	¥3,252	¥2,771

ビデオ		標準価格	48回均等	60回均等
ナショナル	NV-630	¥169,800	¥3,613	¥3,203
〃	NV-330	¥159,800	¥3,170	¥3,015
ナショナル	NV-800	¥289,800	¥6,665	¥5,680
〃	NV-850HD	¥239,800	¥5,238	¥4,700
ソニー	SL-F5	¥169,800	¥3,808	¥3,245
〃	SL-F3	¥145,000	¥3,169	¥2,771
〃	SL-HF77	¥299,000	¥6,877	¥5,860
〃	SL-HF66	¥249,800	¥5,745	¥4,896
ビクター	HR-D225	¥195,000	¥4,373	¥3,726
〃	HR-D120	¥148,000	¥3,310	¥2,842
日立	VT-16	¥173,000	¥3,880	¥3,306

掲載以外の商品もお問い合せ下さい。

全国どこからでも電話1本(無料)で、商品先取り！お支払いは楽々クレジットで胸キュン！です。

豊富なバリエーションの中から自由に組み合わせてください――

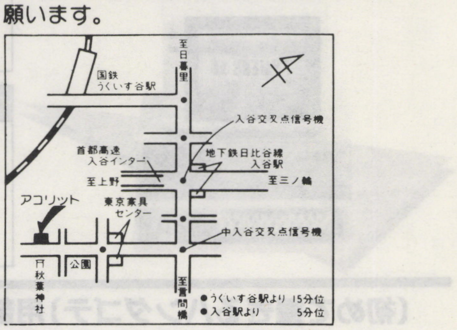
Acolyte
(アコリット)
〒110 東京都台東区北上野2丁目12番12号
営業時間／午前11時～午後7時　年中無休
03(844)3361

●他にオーディオ、ビデオ、パーソナル無線も取り扱っています。クレジットは1～60回までOKです。
●初回金のみ、端数金額処理のため、上記金額よりも少し多くなります。
●契約後の取消は規定の手数料を申し受けます。

お支払いは楽な長期クレジット

48回/60回

●現金の場合はお問い合わせ下さい。クレジットは3,000円以上の組み合わせに願います。



●この他の新製品の御予約も受付けます。

どこまで広がるか
コンピュータライフ。
SOFT新時代にむけて、
いまITEMから‥‥

NEW

disk mate 80E
5インチデュアル
フロッピーディスク

- disk-80シリーズの普及機
- 大きさも一周り小さくなってよりコンパクトに
- リニアシーク機構採用
- このクラスで初めて10万円を切って新登場！

disk mate 80Eシリーズ

- 80E-1 (NEC PC用) …… 108,000円
- 80E-2 (FM-7用) ………… 98,000円
- 80E-3 (X1用) ………… 108,000円
- PC用専用ケーブル ……… 5,000円
- X1用, FM-7用 ………… 12,000円 (インターフェイス・ケーブルセット)

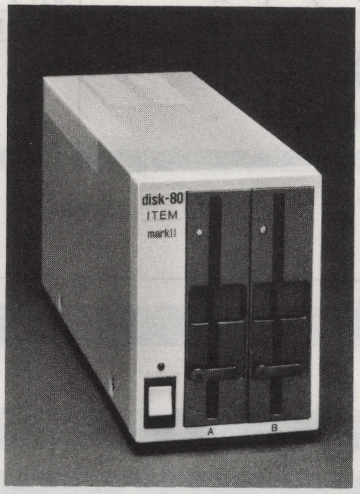
98,000円
(標準仕様)

今日からは Disk!
5インチシングル
フロッピーディスク

disk mate M10シリーズ

- M10-1 (NEC PC用) ……… 69,800円
- M10-2 (FM-7用) ………… 69,800円 (インターフェイス込)
- M10-3 (X1用) ………… 78,000円 (インターフェイス込)
- M10-9 (2台目用) ………… 58,000円
- 専用ケーブル ………… 5,000円

快走！ disk-80mark II
5インチ高速デュアル
フロッピーディスク

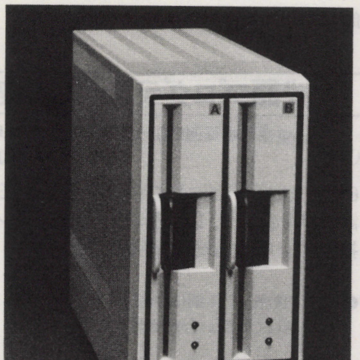


- 高速！そのうえ静寂
- メカ部に定評の松下通工製JA551を採用
- トラック間アクセスタイム6m secを実現

- disk-80P II (NEC PC用) 123,000円
- disk-80F II (FM-8, 7用) …… 128,000円
- disk-80B II (MZ用) …… 138,000円
- 専用ケーブル …………… 5,000円
- FM-8, 7用 I/O ………… 15,000円

123,000円
(disk-80P II本体)

実績No.1 パワフルディスク
8インチデュアル
フロッピーディスク



Disk-PC M160

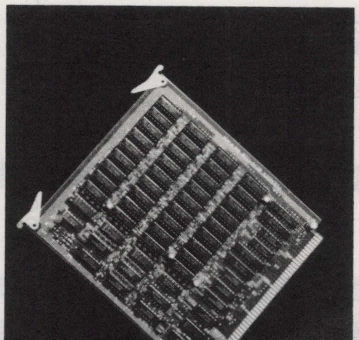
- Disk-PC M160-01 …256,000円 (PC-9801用)
- Disk-PC M160-02 …298,000円 (PC-9801用 CP/M86付)
- Disk-PC M160-03 …296,000円 (PC-8801, PC-8001mk II用)
- Disk-PC M160-04 …266,000円 (PC-9801E, F用。インターフェイス別)

*PC-9801E, Fでご利用の場合は別途インターフェイスが必要です。

256,000円
(専用ケーブル付)

NEW

RAMディスクも夢でない！
PC-9801E, F用 増設メモリーボード M256



- 256KB実装済
- PC-9800シリーズにそのままダイレクトイン
- 最大512KBまでの増設ができます。
- パリティエラーチェック機能付

69,800円
(256Kバイト実装済)

株式会社アイテム
〒251 藤沢市南藤沢8-1-202
TEL 0466-27-1668(代)
FAX 0466-27-2800

- お買い求めは全国マイコンショップで。
- 入手できない方は通販を行なっております。直接当社にご連絡下さい。

*CP/MRはデジタル・リサーチ社の登録商標です。*UCSD Pascal(TM)はカリフォルニア大学理事会の登録商標です。*FLEXRはTSC社の登録商標です。

FM版
COMAS-WPシリーズ

価格据え置きで一新!!（PC版も近日中に一新します）

新WPシリーズには、パソコン・ソフトの限界に絶えず挑戦する当社ならではの新機能がズラリ！
全WPに機能が一目でわかるオーバーレイシートを付属し、マニュアルも全面改訂するなど、扱いやすさもグーンとアップ！

昨年末に予告いたしましたとおり、当社の日本語ワープロ・ソフトWPシリーズが1月中に大幅に機能アップします。
新WPシリーズの機能等は以下のとおりですが、現WPユーザーの皆さまにも若干の手数料で新シリーズと交換させていただきます。

《COMAS-WP 機能一覧》

対応機種		FM-7&8		FM-8	FM-11
FDサイズ		5インチ	5インチ	8インチ	5インチ
製品名		WP-V1	WP-V2	WP-V8	WP-E1
入力	かな漢字短文節変換	○	○	●	○
	ローマ字短文節変換	○	○	●	○
	JISコード変換	○	○	●	○
	•最終使用優先学習機能	○	○	●	○
	横倍角文字	●	●	●	●
	半角英数字	●	●	●	●
	外字処理		●	●	●
	アンダーライン	○	○	●	○
	罫線・枠組機能	○	○	●	○
	センタリング	○	○	●	○
	右詰め・左詰め	○	○	●	○
編集	訂正（上書き方式）	○	○	●	○
	空白挿入	○	○	●	○
	1文字削除	○	○	●	○
	文字列・行挿入	○	○	●	○
	文字列・行削除	○	○	●	○
	文字列・行複写	○	○	●	○
	上段文字複写		○	●	○
	短文登録（MAX.80字）	○	○	●	○
	計算機能		○	●	○
	文字列検索		○	●	○
	文書切り貼り		○	●	○
	文書挿入	○	○	●	○
	1頁の行数設定・変更	○	○	●	○
	左右マージン設定・変更	○	○	●	○
	タブ位置設定・変更	○	○	●	○
	ヘルプ	○	○	●	○
印刷	用紙サイズ指定	○	○	●	○
	単表印刷（ビジネスandMB27410）*	○	○	●	○
	縦書き・横書き指定	○	○	●	○
	袋閉じ印刷	○	○	●	○
	文字間・行間指定	○	○	●	○
	複数部印刷指定	○	○	●	○
	印刷開始頁指定	○	○	●	○
	頁印刷指定	○	○	●	○
	印刷書式の記憶		○	●	○
サポート	文書削除	○	○	●	○
	文書複写	○	○	●	○
	文書初期化	○	○	●	○
	文書圧縮	○	○	●	○
	文書自動圧縮		○		○
	文書一覧表印刷	○	○	●	○
	外字登録，熟語追加		○	●	
価格		￥49,800	￥75,000	￥75,000	￥55,000

*（MB27410でカットシート使用可）

●はプリンターがMP-80Kでは機能しません。

FM版WPユーザーの方で、オーバーレイシートのみをご希望の方は、シート希望の旨とお手持ちのWPの製品名・シリアルナンバーをお書きの上、実費郵送料を同封して当社宛ご請求ください。

ソフト講座のご案内

COMAS-K2活用講座

（開講中）●毎週金曜日1～5P.M./全1回
受講料＝5,000円

K2ユーザーの方を対象として、受講者すべてに実際にマシンを操作していただきながら、専任の講師が指導します。

機械語プログラミング講座

（'83.1/9開講）●毎週2回/全24週（月曜and水曜）
受講料＝60,000円

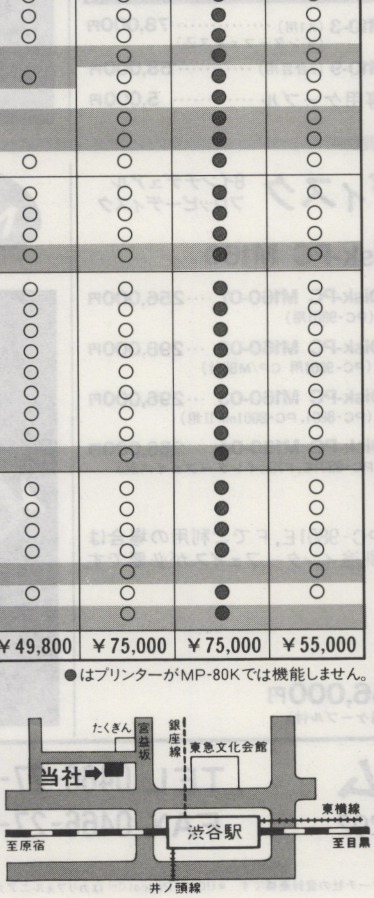
6809アセンブラ言語によるプログラミングを専任の講師が指導します。1人1台ずつ、実際にマシンを操作していただきながら講義を進めていきます。

●「ソフト講座」の詳細につきましては、当社までお電話にてお問い合わせください。

技術者募集

（'84春・新卒者募集中）
会社訪問随時受付

Computer Applications Service
株式会社コマス
パソコン事業部
〒150：東京都渋谷区渋谷1-14-6（平野屋ビル2F）☎(03)407-8893（代）



ジャスコ CVA
COMPUTER VISUAL & AUDIO

NEC PC-9801F

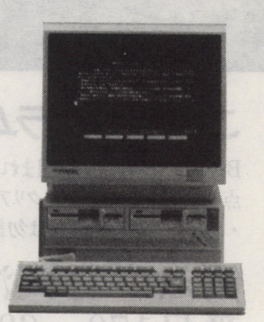
①PC-9801F2 ………………………… 398,000円
PC-KD551 ………………………… 118,000円
合計価格 516,000円

●20回支払(ボーナス併用)例 月々14,200円 (ボ)7.5万×3回
●48回均等支払例 月々12,300円

②PC-9801F1 ………………………… 328,000円
PC-KD551 ………………………… 118,000円
合計価格 446,000円

●20回支払(ボーナス併用)例 月々12,900円 (ボ)6万×3回
●48回均等支技例 月々10,600円

NEC PC-8801mkII



①PC-8801mkIIモデル30(本体) ……………… 275,000円
PC-KD551(14インチカラーディスプレイ)… 118,000円
合計価格 393,000円

●20回支払(ボーナス併用)例 月々11,000円 (ボ)5.5万×3回
●48回均等支払例 月々9,300円

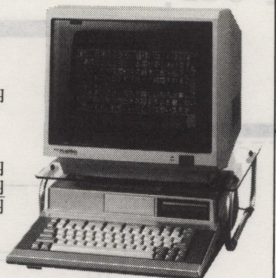
②PC-8801mkIIモデル20(本体) ……………… 225,000円
PC-KD551(14インチカラーディスプレイ)… 118,000円
合計価格 343,000円

●20回支払(ボーナス併用)例 月々10,100円 (ボ)4.5万×3回
●48回均等支払例 月々9,300円

③PC-8801mkIIモデル10(本体) ……………… 168,000円
PC-KD551(14インチカラーディスプレイ) ……… 118,000円
合計価格 286,000円

●20回支払(ボーナス併用)例 月々8,000円 (ボ)4万×3回
●48回均等支払例 月々 6,700円

NEC PC-6601



①PC-6601(本体) ………………………… 143,000円

●20回支払(ボーナス併用)例 月々 4,100円 (ボ)2万×3回

②PC-6601(本体) ………………………… 143,000円
PC-60M43(14インチ15色ディスプレイ) …… 65,800円
合計価格 208,800円

●20回支払(ボーナス併用)例 月々 5,700円 (ボ)3万×3回
●48回均等支払例 月々4,900円

③PC-6601(本体) ………………………… 143,000円
PC-60M43(14インチ15色ディスプレイ) ………… 65,800円
PC-PR401(サーマルプリンタ) ………………… 39,800円
合計価格 248,600円

●20回支払(ボーナス併用)例 月々6,700円 (ボ)3.5万×3回
●48回均等支払例 月々5,800円

SHARP MZ-2200

MZ-2200(本体) ………………………… 128,000円
MZ-1D15(14インチカラーモニター)) ……… 72,000円
MZ-1T02(データレコーダー) ……… 19,800円
合計価格 219,800円

●20回支払(ボーナス併用)例 月々6,200円 (ボ)2.5万×3回
●48回均等支払例 月々4,800円

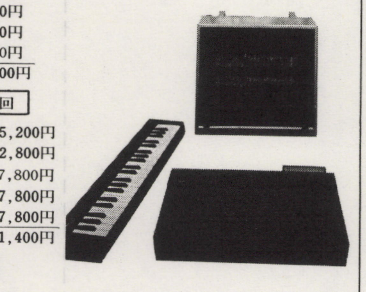
MSX

マイクロソフト社と㈱アスキーが提案した互換性を持ったホームパーソナルコンピュータシステムであり、国内外15社の賛同を得て低価格帯コンピュータの仕様標準。ホビー用からスモールビジネス程度までカバーしている。

	商品名	標準価格	分割支払い例
富士通	FM-X	49,800円	3,200×15回
ナショナル	キングコング	54,800円	3,400×15回
ビクター	HC-6	64,800円	3,200×20回
東芝	HX10D	65,800円	3,100×20回
三洋	MPC-10	74,800円	3,600×20回
日立	HB-H1	62,800円	3,100×20回
三菱	ML-8000	59,800円	3,800×15回
SONY	HB-55	54,800円	3,500×15回
ヤマハ	YIS503	64,800円	3,300×20回

ComputerとAudioがドッキング!?

■YAMAHA ミュージックコンピュータ



① CX-5 (パソコン本体、MSX) 59,800円
YAMAHA YK-01(ミュージックキーボード) 17,800円
YAMAHA SFG-01(FMシンセサイザユニット) 19,800円
YAMAHA RF-01 (家庭用カラーTVアダプタ) 7,800円
合計価格105,200円

20回支払(ボーナス併用)例 月々3,300円 (ボ)1.5万×3回

②①の組み合せ 105,200円
YAMAHA ZPA-01 (プレイカードセット) 12,800円
YAMAHA YRM-11 (FMミュージックマクロ) 7,800円
YAMAHA YRM-12 (FM音色プログラム) 7,800円
YAMAHA YRM-15 (FMミュージックコンポーザ) 7,800円
合計価格141,400円

20回支払(ボーナス併用)例 月々4,500円 (ボ)2万×3回

■VICTOR マイコン制御のビジュアル・オーディオ

ビクター PEACE 90(アンプ、チューナー、プレヤー、デッキ、スピーカー一式) 330,000円
ビクター SEA-G90(フルレコーディングモード、SFAグラフィックイコライザー) 39,800円
ビクター VS-1 (AVグラフィック・シンセサイザー) 79,800円
合計価格 449,600円

20回支払 (ボーナス併用)例 月々14,900円 (ボ)7万×3回

CVA新時代誕生

■SHARP ビデオディスクとコンピューターとの連動

シャープ VP-1000(ビデオディスク・プレーヤー) 129,800円
シャープ VO-100 (ランダム・アクセス・ユニット) 39,200円
シャープ VO-10C1 (接続ケーブル) 9,000円
㊟パソコンのプリンタ命令でコントロールします。 合計価格 178,000円

20回支払(ボーナス併用)例 月々4,700円 (ボ)2.5万×3回

■新映像時代 (X1Cにビデオデッキなどをつけて)

①SHARP CZ-801C (パソコン本体) 119,800円
SHARP CZ-801D (カラーCRT) 99,800円
SHARP CZ-8DT (デジタルテロッパー) 89,800円
ソニー SL-F17 (ビデオデッキ) 132,000円
合計価格 441,400円

20回支払(ボーナス併用)例 月々121,00円 (ボ)6万×3回

②NEC PC-6000mkII (パソコン本体) 84,800円
NEC PC-60M43 (14インチ15色カラーCRT) 65,800円
NEC PC-60M54(スーパーインポーズユニット) 39,800円
NEC VC-709 (ビデオデッキ) 159,800円
合計価格 350,200円

20回支払(ボーナス併用)例 月々9,900円 (ボ) 4.5万×3回

マイコン通信販売
低金利・低価格

お支払方法：クレジットのお支払は ⓐ銀行口座をお持ちの方は自動引落。ⓑ銀行口座をお持ちでない方は、クレジット会社あてに振込にてご送金いただきます。
商品到着後4日以内ならキャンセルできます。但しご使用なさった商品はご遠慮下さい。

お申し込みは☎コレクトコール〈106番で!!〉
★東京(03)246-8948(代表)
＊都内の方はコレクトコールはできません。

■クレジット回数と金利 (ボーナス併用もあります)

回数	1回	3回	6回	10回	15回	20回	24回	30回	36回	48回
手数料	——	1.95%	3.9%	6.5%	9.75%	13%	15.6%	19.5%	23.4%	31.2%

1. 全国無料配達 ★製品即納!!ただし、一部地域は除きます。
2. 低金利クレジット ★金利はグットお得な低金利★回数も自由に選べます。★初回支払いは端数がプラスされます。
3. 頭金なしで商品先どり ★電話1本でO.K!!
4. 完全保証 ★メーカー保証書付です。★取扱い上のトラブルは電話で専門スタッフがお答えします。
5. 日・祭日も受付可 ★受付時間 午前10時～午後7時

創業宝暦8年(1758年) 資本金/98億3,373万円 売上高/6,518億円(昭和58年2月期)

ジャスコ株式会社 CVA事業部
〒103 東京都中央区日本橋本町1−14

スクリーンBASICなら、BASICプロ

スクリーンBASICはBASICプログラミングの画面設計と

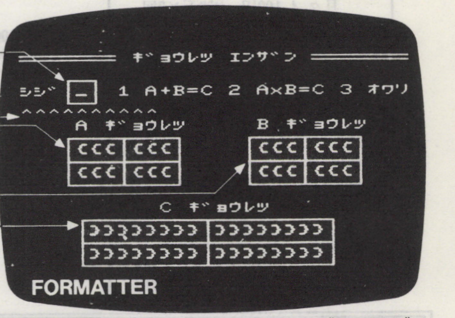
このプログラム・リストをごらんください。

BASICプログラムを組まれる方なら、スクリーンBASICがいかに強力なプログラミングツールであるか良くお解りですね。スクリーンBASICはBASICの欠点とその限界を完全にクリアしました。しかもBASICの持つ優れた計算能力を最大限に活かし、汎用性に結びつけました。スクリーンBASICプログラミング・パターンは、熟練者は勿論、初心者にも非常に解り易く、プログラミング作法が大きく飛躍するまさに画期的なBASICプログラミング用ツールです。

例えば「行列演算」プログラムを組んだ場合、貴方はステートメント番号170～630をプログラミングするだけでOK！

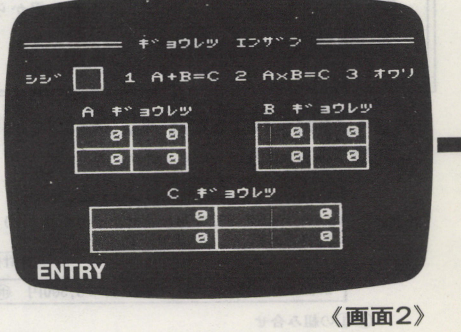
```
100 '[1] スクリーン BASIC OPEN
110 DIM UM$(24):U=&H5300:UZZ=U+3:UMFS=U+6:UHYJ=U+9
120 UBEGIN=U+12:USET=U+15:UEND=U+18:UOPT=U+21
130 U%=1:U%=VARPTR(U%):POKE U,INT(U%/256):POKE U+1,U% MOD 256
140 UZ$="11":U%=VARPTR(UZ$):EXEC UZZ
150 '[2] スクリーン フォーマッター ノ ジッコウ
160 GOSUB 650:EXEC UMFS
170 '[3] センゲン
180 DIM A$(2,2),B$(2,2),C$(2,2)
190 '[4] エントリー ヘンスウ ノ ショキチ セッテイ
200 FOR I=1 TO 2
210   FOR J=1 TO 2
220     A$(I,J)="0":B$(I,J)="0":C$(I,J)="0"
230   NEXT
240 NEXT
250 KBN$="":MSG$=""
260 '[5] ガメン ヒョウジ
270 GOSUB 650:EXEC UHYJ
280 '[6] エントリー
290 EXEC UBEGIN
300 U%=VARPTR(KBN$):EXEC USET
310 U%=VARPTR(MSG$):EXEC USET
320 FOR I=1 TO 2
330   FOR J=1 TO 2:U%=VARPTR(A$(I,J)):EXEC USET:NEXT
340   FOR J=1 TO 2:U%=VARPTR(B$(I,J)):EXEC USET:NEXT
350 NEXT
360 FOR I=1 TO 2
370   FOR J=1 TO 2:U%=VARPTR(C$(I,J)):EXEC USET:NEXT
380 NEXT
390 EXEC UEND
400 '[7] シジ ハンテイ
410 MSG$=""
420 IF KBN$="1" THEN 470
430 IF KBN$="2" THEN 540
440 IF KBN$="3" THEN END
450 MSG$="シジ エラー":GOTO 290
460 '[8] A+B=C ノ ケイサン
470 FOR I=1 TO 2
480   FOR J=1 TO 2
490     C$(I,J)=STR$(VAL(A$(I,J))+VAL(B$(I,J)))
500   NEXT
510 NEXT
520 GOTO 290
530 '[9] AxB=C ノ ケイサン
540 FOR I=1 TO 2
550   FOR J=1 TO 2
560     T=0
570     FOR K=1 TO 2
580       T=T+VAL(A$(I,K))*VAL(B$(K,J))
590     NEXT
600     C$(I,J)=STR$(T)
610   NEXT
620 NEXT
630 GOTO 290
640 '[10] ガメン テイギ サブルーチン
650 UM$( 1)="                                                  "
660 UM$( 2)="                                                  "
670 UM$( 3)="    ════════ ギョウレツ エンザン ════════          "
680 UM$( 4)="                                                  "
690 UM$( 5)="                                                  "
700 UM$( 6)="    シジ |_| 1 A+B=C 2 AxB=C 3 オワリ              "
710 UM$( 7)="                                                  "
720 UM$( 8)="    ^^^^^^^^^^^                                   "
730 UM$( 9)="      A ギョウレツ         B ギョウレツ            "
740 UM$(10)="                                                  "
750 UM$(11)="      |<<<<|<<<<|         |<<<<|<<<<|             "
760 UM$(12)="                                                  "
770 UM$(13)="      |<<<<|<<<<|         |<<<<|<<<<|             "
780 UM$(14)="                                                  "
790 UM$(15)="                                                  "
800 UM$(16)="               C ギョウレツ                        "
810 UM$(17)="                                                  "
820 UM$(18)="       |>>>>>>>>>|>>>>>>>>>|                      "
830 UM$(19)="                                                  "
840 UM$(20)="       |>>>>>>>>>|>>>>>>>>>|                      "
850 UM$(21)="                                                  "
860 UM$(22)="                                                  "
870 UM$(23)="                                                  "
880 UM$(24)="                                                  "
890 UZ$="01":U%=VARPTR(UM$(1)):RETURN
```

作成・修正



《画面1》

表示



《画面2》

こんなに簡単

(1)スクリーンBASIC OPEN

スクリーンBASICの機能モジュール（BASICのステートメントに相当します）と、パラメータの登録を行っています。

- 機能モジュール〔UMFS、UHYJ、UBEGIN、USET、UEND等〕
- パラメータ　〔UM＄（24）、U%、UZ＄〕

(2)スクリーン・フォーマッター ノ ジッコウ

スクリーンBASICには画面フォーマットの作成支援ツールとして、『スクリーン・フォーマッター』が用意されています。このフォーマッターには強力なスクリーン編集機能がありますので、どんな複雑な画面でも直接、CRT上に絵を描く要領で簡単に作成することができます。（画面1参照）ここで作成された画面フォーマットは、そのままステートメント番号650～880の間に自動的に転記されます。

(3)センゲン

この例ではA、B、Cの各行列を配列として取り扱う為、ここで宣言を行っております。

A＄（2,2）…A行列　　B＄（2,2）…B行列

C＄（2,2）…C行列

(4)エントリー ヘンスウ ノ ショキチ セッテイ

画面が最初に表示された時の、各フィールドの値を設定します。

グラミングが大幅に簡単になります。

I/Oコントロール部分を完全にノンプログラミング化しました。

スクリーンBASIC

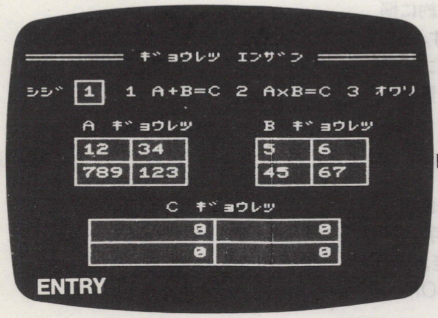
10倍の生産性と100%の汎用性

FM-7用

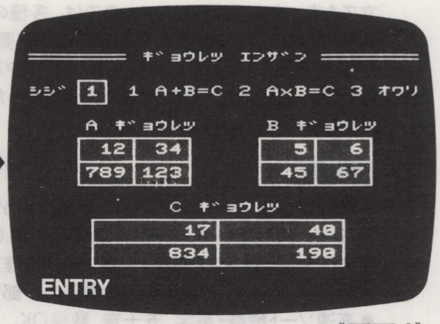
スクリーンBASICはいわば拡張BASIC

BASICプログラミングの画面設計とI/Oコントロール部分を完全にノンプログラミング化したツール・パッケージソフトです。従来のBASICだけでは実現不可能な、より高度で豊富なエディット機能を持つスクリーンBASICは「拡張BASIC」とも言えるソフトです。
スクリーンBASICをお使いになれば、簡易言語では得られない柔軟な汎用ツールパッケージの使いやすさと十分な有用性に必ずご満足いただけます。充実したBASICプログラム・ライブラリーには、ゲームソフトから、ビジネスソフトまで、あらゆる用途に強力な効果を発揮するスクリーンBASICをご活用ください。

データ入力



《画面3》



《画面4》

でスマートなプログラミングが可能です

この例では、A、B、C行列の各要素に"0"を設定し、指示欄とメッセージ欄にスペースを設定しています。(画面2参照)

(5)ガメン ヒョウジ
ステートメント番号650～880の画面イメージがスクリーン上に表示されます。この時、フィールド・インジケータは表示されません。

(6)エントリー
1) 対応付け………フィールドとBASIC上の変数名の対応付けを行います。
2) 表　　示………BASIC上の変数の内容が対応フィールド上に表示されます。(画面2参照)
3) データ入力……表示された画面の入力フィールド上にデータを入力することができます。データ入力時には、エントリーモードでのスクリーン編集機能を利用することができますので、入力、修正が非常に楽に行えます。(画面3参照)
尚、この時にカーソルは、画面フォーマットの作成で"－"と"|"で表現された入力フィールド以外へは移動しませんので、安心してデータ入力を行うことができます。(画面1参照)

(7)シジ ハンテイ
指示欄に入力された値によって、処理の振分けを行っています。

(8)A＋B＝C ノ ケイサン
A行列とB行列に入力された値をもとに、C行列を計算しています。
ステートメント番号520によって計算結果が画面上に表示されます。(画面4参照)

(9)A×B＝C ノ ケイサン
(8)と同様

(10)ガメン テイギ サブルーチン
(2)で説明した通り、UM＄(1)～UM＄(24)の内容はスクリーン・フォーマッターによって書き替えられます。
UZ＄では画面サイズを指定しています。この値を変更することによって次の4種類の画面サイズを選択することが出来ます。

- UZ＄＝"00"……40字19行
- UZ＄＝"01"……40字24行
- UZ＄＝"10"……80字19行
- UZ＄＝"11"……80字24行

スクリーンBASICの機能

- **フォーマッター**: イメージで画面を作成する画面設計ツール
- **エントリーコントローラー**: データI/O及び、入力時のカーソルコントロールや編集等を行う

フィールドインジケーター

フィールドの種類	通常	右詰
入出力フィールド	＿	|
出力専用フィールド	＾	|

オプション機能

- カラー処理の指定
- ハードコピーのモード指定

製品内容

- ●マニュアル　解説編・操作編・付録 B5版 61ページ
- ●カセットテープ　スクリーンBASIC、プログラミング・パターン(2種)　サンプル・プログラム集（グラフ・計算書・計算する黒板・マスターマインドゲーム・BASIC関数計算）
- ●価格　9,800円

●流通販売
ソフトウェア・インターナショナル株式会社
〒150 東京都渋谷区神宮前6丁目12番20号東洋ビル9F
●製造
株式会社ユーザーズソフト
〒158 東京都世田谷区奥沢6-33-9内海ビル301 TEL(03)704-0821(代)
※お問い合せは、下記インフォメーション・センターまで

インフォメーション・センター
TEL.(03)486-7155

● 全国有名マイコンショップでお求め下さい。

FM-7/8

FMS

FILE-MANAGEMENT-SYSTEM

今までのデータベースとココが違います!!

今までのカード型データベースに、あきたらないあなたへ
情報処理にムダな時間をついやしていませんか?
FMSは、そんなあなたに、ほんのわずかな時間でお役に立てます。今までのデータベースでは、各種の制約に悩まされましたが、FMSでは、そんな悩みは不要です。メモ帳感覚で使っていただけます。それにディスク1枚に2,048件ものデータ管理ができ、今まで出来なかった大量データ処理も可能になりました。

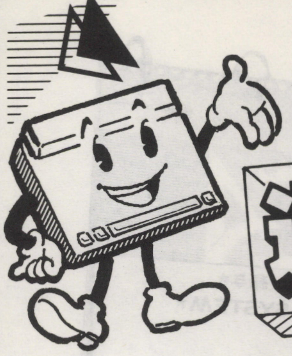
〔特長〕

- ●画面は完全フリーフォーマット・書式設定が、自由自在。
- ●1レコード最大32ページ・文字数では約56,000字。
- ●データ件数最大2,048件・大量データ処理可能。
- ●強力検索機能・多重検索、文字列の一部でもOK。
- ●高速ソート機能・英字、五十音、数値OK。
- ●項目設定　自由自在。(設定制約なし)
- ●レポート機能・計算機能を完全サポート。
- ●プリンターへの印字は書式自由。

マニュアル・特製バインダー付き
5インチ ¥29,800
マニュアル別売り致します。

Policy 株式会社ポリシー
〒542 大阪市南区南炭屋町43 アベニュー心斎橋 TEL 06(211)8987

■皆様からの御意見・御要望をお待ちしております。
■お求めは、有名パソコンショップか、現金書留で住所、氏名、商品名、機種名明記の上、左記までお送り下さい。尚、送料として350円加算して下さい。



MDBA 通信添削講座

● 自由な時間にプログラミング，いかがですか？

パソコンを買ったのに，プログラムを習うチャンスがなく，市販ソフトだけでがまんしている方，自習書などで勉強しているけれども，よくわからない方，歓迎いたします。

● 実技中心の講座です

プログラミング技術は，何本も何本もプログラムを作ることにより上達する技術です。当講座は，出来るだけプログラミングをして頂くよう，ワークカセットを用意し，当社に送られてきたプログラムは，プリンタからリストを出力し，添削して返送する出題形式等，工夫をこらしたカリキュラムになっています。

● 幅広い学習が期待できます

当社はゲームソフトから簡易言語，データベース，コンピュータ・グラフィックス・ソフトまで開発してきたシステムハウスですから，幅広いプログラミング技術が学べます。

● プログラム作りには良い道具が必要です

ご入会の方には，プロが使っている，使い易く，保管し易いMDBA特製仕様書類及びバインダー，テンプレートを使って頂きます。良いプログラムを作るには，良い道具が必要です。あなたの部屋も明日からはシステム開発室です。

対象機種 ①PC-8001mkII，PC-8001 ②FM-7 ③X-1

（PC-6001mkII，MSX用も開講準備中）
※機種別の講座ですから，あなたのパソコンにあわせた学習が行なえます。

講座

①ビギナー講座	初めての方，基礎から始めたい方を対象。
②初級講座	FORループが使える方を対象。
③中級講座	開講準備中。

バグカード　システム

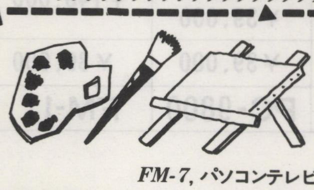
自作プログラムでどうしてもわからないバグにお答えいたします。

※実技中心の講座のため，ご入会はパソコンをお持ちの方に限らせて頂きます。

資料希望の方はハガキに〒，住所，氏名，TEL，機種名を記入の上，当社宛にお送り下さい。早速カタログをお送りいたします。

マイクロ データ ベース アソシエイツ
〒194-01 東京都町田市鶴川2-1-8-205
☎(0427)35-1600

MDBA *software library*



ART CREATOR

FM-7, パソコンテレビ *X-1*用　**3,800円**

機能説明

コンピュータ・グラフィックスの世界で遊んでみませんか。ART CREATOR は、コンピュータで絵を描くためのソフトウェアです。
紙に絵を描くのと同じように、骨組みは太ふでを使って太く、大まかに描き、細かいところは細く、ゆっくり描けます。使えるコマンドは15種類あります。色は単色の他に、カラーパターン自動生成機能により自由にカラーパターンをつくれます。また、ユーザー定義キャラクタをカラーパターンとして使えますので、3万色以上のカラーパターンを表現できます。

1. 四角、直線、円、楕円、多角形を簡単に描けます。
2. ぼかし、かすれ等を描くブラシ機能があります。
3. ジョイスティックが使えます。
4. 640、320両モードで描画できます。
5. コーザ定義キャラクタを定義できます。
6. PALET 変更、カセットへの記録 他、機能を満載しています。

FUMW13002
SIMW16003

※FM-7用には3，4，5の機能はありません。

■ **BATTLE ARMY** FM-7，PC-8001mkII，X-1用　2,800円
キャラクターが美しい思考型ウォーゲームです。思考レベルは3段階，作戦によってコンピュータ側の判断も変化します。

■ **エコノミーパック PART-1** MZ-2000，-80B用　4,800円
住所録と簡易言語の汎用集計表がパックになったお買得なビジネスパックです。

■ **MUSIC EDITOR** MZ-2000，-80B用　2,800円
オルガンをひく感覚でMUSIC文を作成します。削除，追加，カセットのLOAD・SAVEもOKです。

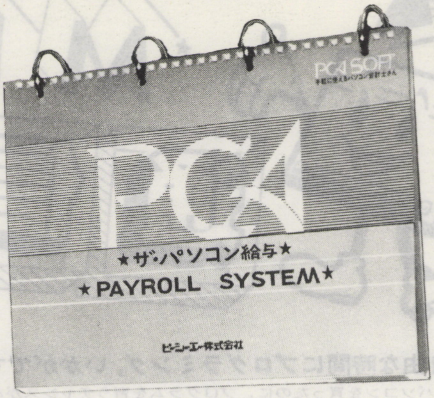
■ **SKY DIVING** MZ-2000，-80B用　2,800円
海面に浮かぶリモート船を操作しながら降り立つゲームです。カモメやサメをよけて着地に成功すると，サメとの死闘が待っています。

■ **4人ポーカー** FM-7，PC-8001mkII，X-1，MZ-2000，-80B用　2,800円
コンピュータがカード交換からかけ点まで全てを判断します。反応が人間に近くおもしろいゲームです。

漢字版

PCA SOFT

手軽に活用。パソコンOA



ザ・パソコン会計

監修 公認会計士　吉村成弘

〈10大特長〉

❶コンピュータを知らなくても家計簿程度の知識で誰にも使えます。
❷使用科目名がイージーオーダー式でどんな科目名も設定できます。
❸現在ご使用中の伝票や出納帳がそのまま使えます。
❹日常よく使われる主要48科目がワンタッチ入力できます。
❺入出金データの入力は自動仕訳方式による等、使いやすく、正確で、しかも主要部分はマシン語処理により、トータルパフォーマンスを追求しています。
❻データの訂正、追加、削除も簡単にできます。
❼日付単位の期間指定、科目指定により、いつでも必要な資料が作成できます。
❽図解入りのカラー印刷で、スタンド式の分かりやすい操作説明書がついています。
❾1000社以上の豊富な使用実績があります。
❿対応機種が豊富です。
漢字版　PC-9800, JB-3000(FM-11, SP-200開発中)
カナ版　FM-7, 8, 11

〈機能〉

●主勘定科目(240)漢字７文字●補助科目(240)漢字７文字●摘要文漢字14文字●摘要文コード登録(200)　●合計金額(1千億円未満)　●月末期末自動更新●他
＊詳しくは「ザ・パソコン会計」公認会計士吉村成弘監修(第一法規出版)￥1,600を参照ください。

〈ソフト価格〉

●システムＡ(元帳, 試算表, 決算書など10表)　￥98,000
●システムＢ(Ａ＋要約日計表, 補助簿, 手形管理など14表)　￥148,000
●システムＣ(Ｂ＋部門管理, 各種分析など21表)　￥198,000

近々発売予定

PC-9800用　販売売掛管理システム（名古屋開発）　￥98,000
FMシリーズ用　利益計画管理システム（大阪開発）　￥98,000

その他ソフト

MZ-80B, 2000, 2200用　高級財務会計、売掛金管理、買掛金管理、給与計算、顧客管理　MY・REPO-180
高級比較棒グラフ、減価償却計算、年末調整など。

ザ・パソコン給与

監修 公認会計士　小川哲英

〈10大特長〉

❶コンピュータを知らなくても誰にでも簡単に使えます。
❷就労基準、支給、控除項目などイージーオーダ式でお客様の給与体系に合せて使えます。
❸給与計算から年末調整まで連結でも単独でも使えます。
❹固定データと変更データが個人単位に画面で一覧できます。
❺月々は残業時間などの変動データを入力すれば煩雑な計算を全てパソコンが処理します。
❻データの訂正、追加、削除も簡単にできます。
❼個人別やグループ別に計算が自由にできます。
❽住民税の納付先一覧表も作成します。
❾賞与査定のための試算表が作成できます。
❿ザ・パソコン給与IIは各種人事管理が作成できます。

〈機能〉

●対象人数(200)　●支給項目数(14)漢字６文字●控除項目数(13)漢字６文字●個人名文字数(漢字10文字)●部門数(9)●対象者(時給、日給、月給)●所得税(甲、乙欄)●雇用保険(有、無)●有給休暇管理●扶養数●他

〈ソフト価格〉

	給与Ｉ	給与II
●給与計算編(6表)	￥59,000	￥148,000
●賞与計算編(6表)	￥39,000	
●年末調整編(5表)	￥39,000	￥39,000
対応機種	PC-9800	FM-11

ご来社テスト自由、資料請求は〒200分を同封の上資料請求券を添えてお申し込み下さい。

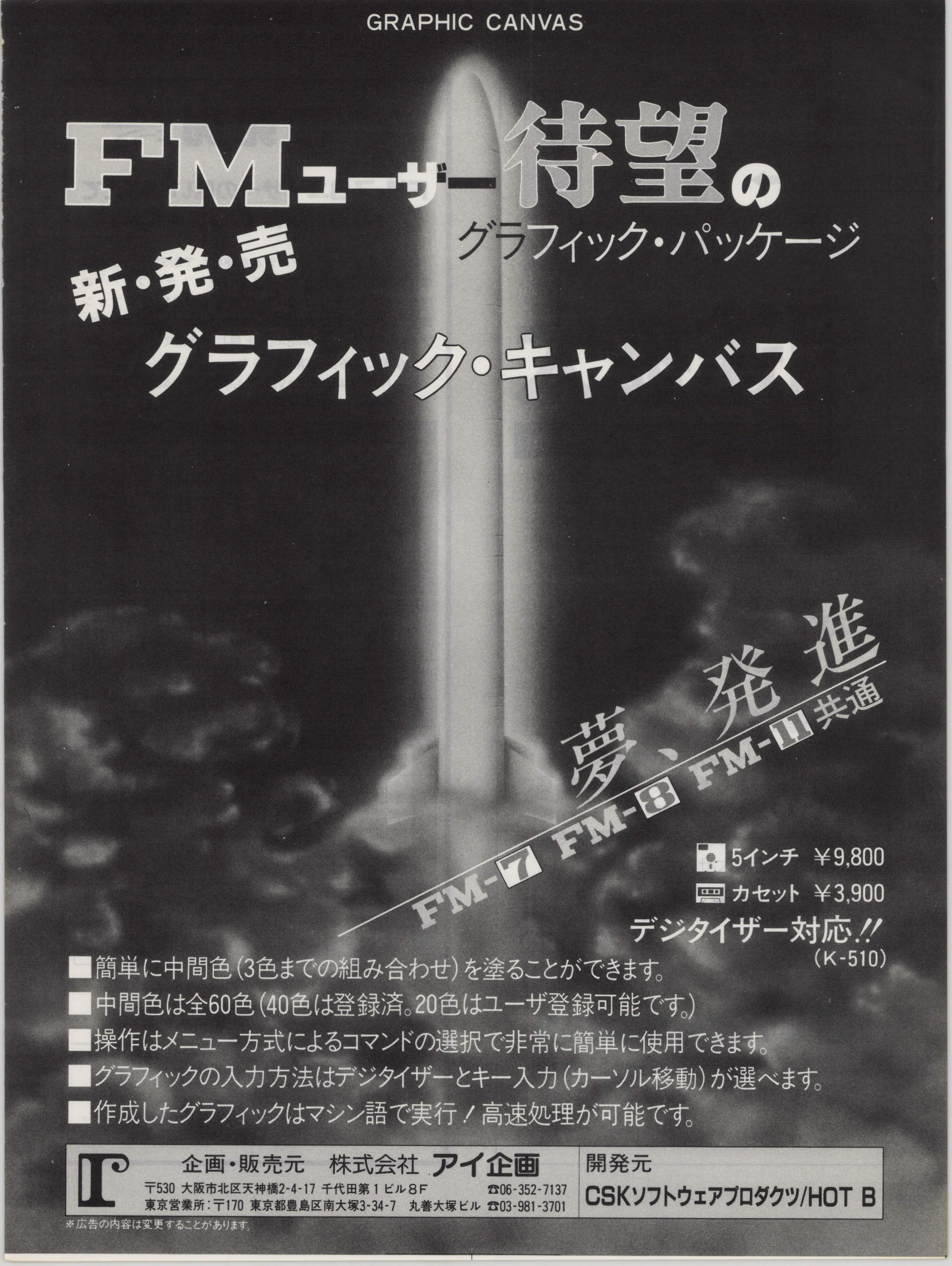
販売代理店募集中

お問い合せは電話かハガキで
担当：三島、岡田まで

ピーシーエー株式会社

〒160 東京都新宿区西新宿７－７－33(新銘ビル新館２Ｆ)　TEL 03(368)9631㈹
〒530 大阪市北区中崎西２丁目２－１(東梅田八千代ビル)　TEL 06(315)8637㈹
〒451 名古屋市西区菊井２丁目６－７　TEL 052(565)0406㈹

資料請求券
Oh！FM


GRAPHIC CANVAS
FMユーザー待望の
グラフィック・パッケージ
新・発・売
グラフィック・キャンバス
夢、発進
FM-7 FM-8 FM-11共通
5インチ ￥9,800
カセット ￥3,900
デジタイザー対応!!
(K-510)
■簡単に中間色(3色までの組み合わせ)を塗ることができます。
■中間色は全60色(40色は登録済。20色はユーザ登録可能です。)
■操作はメニュー方式によるコマンドの選択で非常に簡単に使用できます。
■グラフィックの入力方法はデジタイザーとキー入力(カーソル移動)が選べます。
■作成したグラフィックはマシン語で実行! 高速処理が可能です。
企画・販売元 株式会社 アイ企画
〒530 大阪市北区天神橋2-4-17 千代田第1ビル8F ☎06-352-7137
東京営業所:〒170 東京都豊島区南大塚3-34-7 丸善大塚ビル ☎03-981-3701
開発元
CSKソフトウェアプロダクツ/HOT B
※広告の内容は変更することがあります。

項目設定 → データー入力 → 多重計算印刷

用途拡大

プログラミングレスソフト OAソフト-01

お手持ちのパソコンは
充分活躍していますか。
OA-01は
広範にわたりパソコンの偉力を
引き出します。

主な機能

★マルチソート、マルチ計算機能。
★データー入力も表形式、1画面入力も可。
★プリンター出力も書式自在。
★指定データーによる集計及び計算も可。
★小形、中計、大計等も可。

FM-7,11用 OAソフト-01
¥76,000

ダイナミック株式会社

千葉県成田市東和田355番地
〒286 TEL 0476(24)2423(代)
担当 二瓶(にへい)

新発売

FM-7,8ユーザーの願いに応えて ゲーム用NEW DOS遂に完成

このDOSは、DISKシステムを使用してDISKコードの領域(＄7000～＄7FFF)を占める、カセットテープモードのマシン語プログラム、あるいはゲーム用(BASIC＋マシン語)プログラムをDISKにて、SAVE/LOADできるものです。フリーエリアは、ROMBASICと同じに使用できます。

使用できるDISK用コマンド――
RUN、SAVE、SAVEM、LOAD、LOADM、FILES
NAME、KILL、DSKIN I 〰〰はFM-8のみ

FMシリーズ用 メディア：5′ディスケット

商品番号	商品名	内容	機種	定価
F1	NEW DOS	上記参照	FM-7、8、	F1＋F2 セット ¥10,000
F2	DOSモード DISK BASIC	DOS(All RAM)モードのDISK BASICを作成するプログラム		
F3	6809アセンブラ (マシン語)	BASICモードで、RAM又はDISKのソースファイルをアセンブルする。	FM-7、8、11	¥8,000
F4	6809逆アセンブラ (マシン語)	2パス方式でラベル付、上記アセンブラのソースファイルとして使用可。データエリア指定可	FM-7、8、11	¥8,000
F5	DISKユーティリティ (マシン語)	ファイルコピー、VOL.コピー、マシン語ファイルのスタート/エンドアドレス表示等のコマンド有	FM-7、8、11	¥6,000
F6	DISKエディター (BASIC)	1セクタリード/ライト、アスキーチェンジ、16進チェンジ(スクリーンエディタ)、トラックコピー等	FM-7、8、11	¥6,000
F7	ファイル変換 (BASIC)	F BASIC↔CP/M両方で使用可能にFAT、ディレクトリを変換する。	FM-7、8、11	¥6,000
F8	6809デバッガ (マシン語)	1パスアセンブラ、逆アセンブラ、スクリーンエディタ式メモリチェンジ、ブレイクポイントセットトレース(レジスタ及逆アセ表示)等コマンド有	FM-7、8、11	ディスク版 ¥12,000 テープ版 ¥10,000
F9	ファイルコピー (BASIC)	PC-8000シリーズのBASIC用DISKをF-BASIC用に変換し、コピーする	FM-7、8、11	¥ 6,000

CP/M用 ※CP/Mはデジタルリサーチ社の登録商標です

商品番号	商品名	内容	機種	定価
C1	Z-80アセンブラ ZAS. COM	ASM. COMと使用方法が同じで、ザイログ型式のニーモニックをアセンブルする。	FM-7、8、11 PC1W、2W パソピア FP-1000 MBC-100	C1＋C2 セット ¥12,000
C2	Z-80逆アセンブラ ZDA. COM	2パス方式でラベル付、ソース、プリントファイルをDISKに作成する。		
C3	ファイル転送 TFR. COM	CP/Mマシン間でRDR：/PUN：を使用してすべてのファイル型式の転送をする。		¥6,000
C5	フォーマット変換 YOU. COM	各社のパソコンのDISKをリード/ライトできる。PC1W、2W用、FM-8、パソピア、MBC-100、M.D.IF800. その他	FM-7、8、11 PC-2W、QC-10，IF-800 パソピア	¥16,000
C6	6809デバッガ DEBUG.COM	F8(6809デバッガ)のFM-CP/M-80バージョン	FM-7、8、11	¥12,000
C7	ラムディスク RAMDISK.COM	上位64KバイトとVRAM192Kバイトをディスクドライブの1台として使用可能にする。SUBMIT文の使用可(バージョンアップ)	FM-11	¥16,000
C8	ディスクエデッタ DE.COM	F6(DISKエデッター)のCP/M-80バージョン	FM-7、8、11	開発中
C9	ラムディスク CP/M-86バージョン	拡張RAMをディスクドライブとして使用。128Kより768KBまで可能。＊新製品＊	FM-11	¥16,000

(F8 6809デバッガのテープ版はFM-7用のみ)

御注文は、販売元へ、商品番号、商品名、機種名を明記の上、現金書留でお願いします。(送料は無料です。)

販売元 **カーネル**
〒130 東京都墨田区横川3-1-5 石野ビル201
TEL 03―624―4921
※情報処理システム研究所の社名，住所，電話番号が変わりました。

開発 **橋口技研**

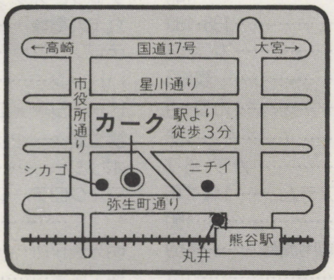
埼玉・熊谷

パソコンショップ CARC〔カーク〕

パソコンの真価はソフトで決まる！
ソフト・専門図書充実のパソコン専門店

営業時間　AM10：00〜PM８：00
定休日　毎火曜日

取扱いメーカー
NEC
シャープ
富士通
東芝
カシオ
アップル
エプソン
ETC.



テープ版ワープロの決定版

9月1日全国一斉発売

日本語簡易ワードプロセッサー 書あ句（かく）

¥4,900

MODEL　FUJITSU　FM-7／FM-8　要本体漢字ROM
適合プリンター EPSON RP-80, FP-80, MP-80TYPEIII

特徴
- メモリをフルに活用したテープ版では最高の機能と操作性
- スクリーンエディター方式で文章編集容易
- 一画面40桁×10行、400文字表示
- 漢字の呼出しはカナ２文字まで指定可
- 縦書き・横書き・改行間隔２種指定可
- この機能でこの低価格

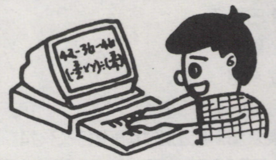
本商品は全国マイコンショップ又は直接当社で（現金書留・送料無料）お求め下さい。販売代理店/㈱日本ソフトバンク・誠光堂書籍㈱

コンピューター応用開発株式会社　TEL0485-25-4432

〒360　埼玉県熊谷市弥生町２－47

ENJOY STUDY!

勉強時間が楽しくなった!!



PERFECT MATHEMATICS

高校数学シリーズ　全50巻

きめこまやかな学習、豊富な例題。グラフィックを使った適切な説明で君の能力を飛躍的にUP！ 6巻発売中

数 I	二次不等式	数 I	点と直線
数 I	二次関数	数 I	円と領域
数 I	分数関数、無理関数	数 I	三角比

■価格／各巻4,000円（59年3月全巻完成）
■機種／PC-6001（MKII可）・FM-7・8

EXPERT HISTORY・GEOGRAPHY

中学・高校社会科シリーズ

長期間に渡るベストセラー！

中学・高校生向	中学・高校生向	中学・高校生向	中学・高校生向
日本史	世界史	日本地理	世界地理

■価格／各巻3,000円　■機種／PC-6001（MKII可）・PC-8001（MKII可）・FM-7・8・MZ-700・1200・X-1

EXPERT MATHEMATICS

中学数学シリーズ　全36巻

10年間の数研塾のノウハウから生まれた親切設計のすごい学習ソフト13巻発売中。

中 1	整数 1	約数・倍数・素因数分解
中 1	整数 2	最大公約数・最小公倍数・文章問題
中 1	正の数・負の数	数直線・加減法・乗除法・混合問題
中 1	文字と式	文字式・式の値・式の計算
中 1	一次方程式 1	等式の性質・一次方程式の解法
中 1	一次方程式 2	文章問題の解法
中 2	式の計算	式の加減・乗除・応用問題
中 2	不等式 1	不等式の解法
中 2	不等式 2	連立不等式の解法
中 2	連立方程式 1	加減法・代入法・（　）の問題・分数の問題
中 3	式の計算 1	展開と乗法公式
中 3	式の計算 2	因数分解とその応用
中 3	平方根	√ の計算と応用

■価格／各巻3,000円（11月全巻完成）
■機種／PC-6001（MKII可）、PC-8001（MKII可）、FM-7・8、MZ-700・1200

EXPERT ENGLISH

中学英語・高校受験シリーズ　全20巻

最新の入試問題を厳選して1200問に編集!!

	第1巻	第2巻
1	整序英作文シリーズ（Bランク）	整序英作文シリーズ（Aランク）
2	語句の定義シリーズ（Bランク）	語句の定義シリーズ（Aランク）
3	適語補充シリーズ（比較構文・主語転換他）	適語補充シリーズ（受動態・能動態・関係代名詞）
4	適語補充シリーズ（接続詞・連語・現在完了形）	適語補充シリーズ（同意異義語）
5	相関語句シリーズ（反意語・同意語・内容関連語）	相関語句シリーズ（語尾変化・派生語）
6	相関語句シリーズ（動詞・形容詞の語尾変化A）	相関語句シリーズ（動詞・形容詞の語尾変化B）
7	全文書きかえシリーズ（関係代名詞A）	全文書きかえシリーズ（関係代名詞B）
8	全文書きかえシリーズ（不定詞・間接疑問文A）	全文書きかえシリーズ（不定詞・間接疑問文B）
9	前置詞シリーズ（at, by, for, from）	前置詞シリーズ（in, of, on, to, with）
10	英作文シリーズ（自由英作文）	英作文シリーズ（和文英訳文）

■装丁／BOOK型2点セット（マニュアル付）　■価格／各5,800円
■機種／PC-6001（MKII可）・（PC-8001・X-1・FM-7・8近日発売!!）

EXPERT ENGLISH

中学英作文シリーズ　全18巻

実像が光るシリーズ、グレードUPして全巻新発売

学年	Part	第 1 巻	第 2 巻
中 1	1	Be動詞・What・Who	have文・命令形
中 1	2	I・You文・複数形・Which	一般動詞・When, How
中 1	3	進行形・There・Can	問題コース付1年英単語辞典
中 2	1	過去形	比較形
中 2	2	未来形	不定詞・受動態
中 2	3	2年総復習	問題コース付2年英単語辞典
中 3	1	動名詞	現在完了形
中 3	2	現在分詞・過去分詞	関係代名詞
中 3	3	中学総復習	問題コース付3年英単語辞典

■装丁／BOOK型2巻セット（マニュアル付）　■価格　各5,800円
■機種／PC-6001（MKII可）、PC-8001（MKII可）、FM-7・8（X-1用近日発売）

株式会社 数研塾

開発事業部

〒411 静岡県三島市一番町15-32
芹沢ビル4F　TEL（0559）75-1534

※振込先：
駿河銀行三島駅前支店　606912（送料不要）

代理店募集中！

◈FMのハードやソフトに詳しい人で，原稿の書ける方を募集しています。電話のうえ，実力評価用の作品をご持参ください。また地方の方は，簡単な自己紹介，マイコン歴，得意な分野などの資料とともに，作品をご郵送ください。グループ応募も歓迎いたします。

〒102 千代田区四番町2-1 ㈱日本ソフトバンク出版部
Oh!FM編集室 松岡 ☎03(261)4095

◈本誌の内容に関して，電話による質問を受け付けます。

☎03(265)5789 受付時間 16:00〜18:00

質問はバグ情報についてのみ受け付けます。機械の取り扱いなどについてのお問い合わせはご遠慮ください。

編集後記

■Oh!FMのニューフェイスです。住所録いかがでしたか。なかなか使いやすいでしょ♡ これからもがんばりますので，万一バグがあってもあたたかい目で見守ってやってください。 (Y)

■巷ではクリスマスがあるそうですが，まったくうちの大学ときたら，ほんの小さな幸せさえも奪ってしまう鬼のようなところです。年末まで試験だなんて，僕はもういやだ。早くお酒が飲みたい。いや，原稿が書きたい。ホントです，嘘じゃありません！(GO)

■製作記事を楽しみにしている読者へ，実は編集部でも最近，TTL I/C 74LS××というなじみのものが，買えなくて困っています。あっても○百円という値段です。以前数十円で買えたのがと思うと，なかなか買えません。ジャンクの基板からはずしたり，使わなくなった基板からはずしていますが，早く正常な状態に戻ってほしいものです。 (小)

■みなさん，アンケートはがきどうもありがとうございます。自分が描いた表紙がほめられると，お世辞でもうれしいものです。

今月の表紙は私の仲間に描いてもらいました。いよいようれしいバレンタインの季節ですね。冬の街角で寒そうに，あなたが通るのを待っている女の子がいるじゃないですか。この表紙は，バレンタインデーに胸を躍らせているみなさんに捧げます。GOOD LUCK TO YOU! (あ)

■また新しい年がやって来ました，Oh!FMの月刊化などの行事(?)も控え，編集部も昨年に増して忙しくなりそうです。

ところで，暮れから新年にかけてより，3月から4月の方が生活の変化が大きいのに，この季節に大きな区切りを感じるのは何故なのでしょうか。きっとこの習慣が小さい頃からしみついてしまっているからなのでしょう。 (ゑ)

■私は年賀状がきらいだ。まだ年も明けぬうちから「あけましておめでとう。今年もよろしく」なんて情けない。別に世話になったわけでもないのに「昨年はお世話になりました」などとしらじらしい。本物の年賀状とは，年が明けてから書くものなのだ。てなわけで今は1月。筆不精な私は，この後記を年賀状がわりにするのです。「あけましておめでとう。今年もよろしく」 (妙)

■本誌の編集人は物資調達係でして，読者プレゼント用に各種とりそろえたり，原稿作成に必要なハード，ソフトを集めてくるのが本業です。で，今回，Oh!FMの名入りシャープペンシルを用意したところ，担当のM氏より「何これ！ 安っぽくてイヤ！」とのこと。日本政府に先んじて行政改革を遂行してるわけじゃないけど，予算っていうものがありましてね。併せて読者の皆様にもお詫び申し上げますが，せっかくのモノですのでご愛用ください。(他誌では評判いいんですよ，念のため) (T)

■今号の特集はいかがでしたか。ディスクなんか持っていない，関係ないよ，なんて言わずに読んでください。パソコンを持っているからにはディスクも持っている，という時代が必ず来ます。次号月刊第1号は，バグとり特集とROM解析です。お楽しみに。なお，編集長推奨シャープペンは，読者プレゼント用としてありがたく使わせていただきます。とても格調の高い品物です。どしどしご応募ください。 (M)

●編集 松岡真理 渡辺妙子
●技術 松田辰夫 小林初雄 徳永 聡
●海外協力 HONG·LIANG·LU IAN·ALLEN RONALD·BILLINGS HISAYOSHI·MIKAMI
●協力スタッフ 菊池蓉子 木下淳博 鶴岡哲明 志沢雅彦 末安直樹 長沼孝仁 西村義孝 平野 剛 古田 昌 本郷誠司 米村英明 林 剛 正
●写真 浜崎 昭 杉山和美
●レイアウト オメガ社
●表紙 Tea Spoon 立花裕一

●広告索引● 五十音順

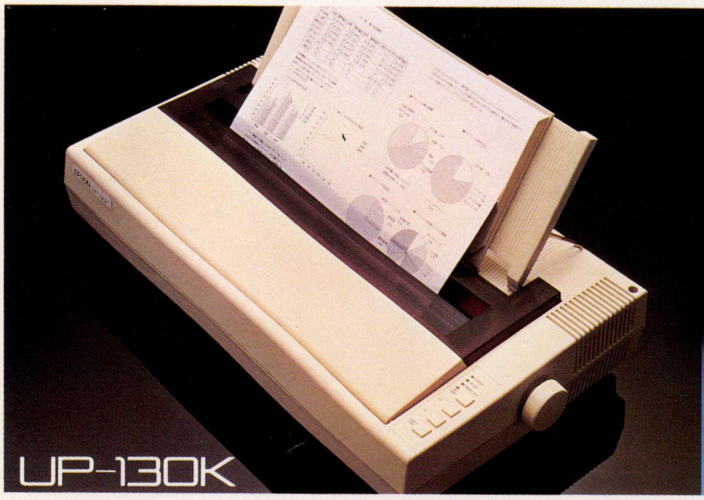
アーバン電子	102	数研塾	159
アイ企画	157	鈴木技研	136
アイテム	149	ソフトウエアインターナショナル	152·153
アコリット	148	ソフトマート	143
亜土電子工業	147	ダイナミック	158
アンプルソフトウェア	142	チャンピオンソフト	カラー12
エイト電気	146	九十九電機	137
エス・ビー・シー	138·139	テクノソフト	141
エニックス	カラー8	ハドソン	カラー10·11
エプソン	表紙3	ピー・シー・エー	156
エム・エー・シー	カラー9	富士通	表紙2·4，カラー1·2·3
カーネル	158	ポリシー	154
関東電子	カラー6·7	マイクロデータベースアソシエイツ	155
光栄マイコンシステム	144·145	明光製作所	140
コマス	150	理工産業	64
コンピューター応用開発	159		
ジャスコ	151		

隔月刊 Oh!FM 第7号
●昭和59年2月1日発行
●発行人／孫 正義
●編集人／田鎖洋治郎
●発行元／株式会社日本ソフトバンク
本社／〒102 東京都千代田区九段南2-3-14 靖国九段南ビル2F
☎03(263)3690 Fax.03(263)3660
出版部／〒102 東京都千代田区四番町2-1 ☎03(261)4095
大阪支店／〒542 大阪市南区難波千日前5-19 河原センタービル
☎06(644)0191 Fax.06(644)0160
●印刷 図書印刷㈱ ☎03(453)2550
●海外送付ご希望の方は下記へお問い合せください。
日本IPS株式会社
〒101 東京都千代田区神田小川町3-5
☎03(291)2632(直)
©1984 JAPAN SOFT BANK 雑誌02199-2
★本誌からの無断転載を禁じます。

EPSON

ペリフェラルはエプソン

多機能漢字プリンタUP-130K、高機能・低価格ターミナルフロッピーTF-10。ユーザーニーズに応えて新登場。(12月発売)

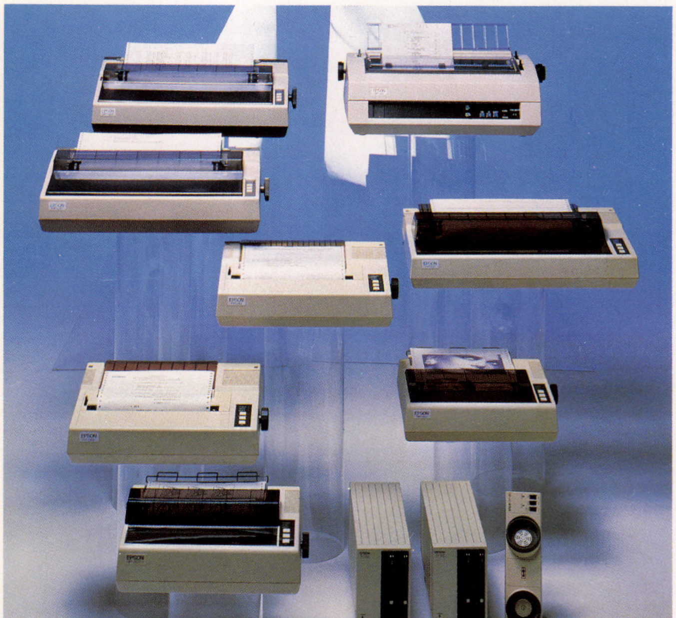


漢字プリンタMP-130Kを大幅にパワーアップした136桁24ピン漢字プリンタUP-130K。●普通文字200cps、漢字45cps印字速度を飛躍的にアップ。印字方向は双方向最短距離印字(ロジカルシーキング)。●ワンタッチで印字開始位置まで用紙を自動的にセットするオートローディング機能などカット用紙の取り扱いが簡単。●ソフトウェアコントロールコードは従来のMP-130Kのコントロール体系をカバーした上で、さらに豊富な機能を追加。●漢字第1水準(4,096文字種)を搭載した機種と、漢字第2水準(3,185文字種)までを搭載した機種の2機種を用意。また、第2水準漢字ROMもオプション販売。

■印字モード：漢字・普通文字・LQ文字(レタークォリティー)・プロポーショナル印字■印字方式：インパクトドットマトリクス(24ピン)■印字方向：漢字・普通文字・LQ文字・ビットイメージ/双方向最短距離印字(ロジカルシーキング)単方向指定時のみ、左→右の単方向印字■文字種類：JIS第1水準準拠約4,000＋160文字種(記号等を含む)、JIS第2水準準拠約3,200文字種■文字構成：漢字24×24ドットマトリクス、普通文字9×17ドットマトリクス、LQ文字15×17ドットマトリクス、プロポーショナル印字37×17ドットマトリクス■用紙：単票用紙　紙幅/182～364mm・紙厚/0.06～0.1mm・連量/45～70kg、ファンフォールド紙(トラクタ使用時)　紙幅/101～406mm・紙厚/MAX0.3mm■コピー：ファンフォールド紙オリジナル1枚＋3枚■印字速度：漢字約45cps、普通文字200cps■外形寸法：604W×363D×130Hmm■重量：14kg■価格：¥290,000(JIS漢字第1水準)¥310,000(JIS漢字第2水準)〈オプション〉シリアルユニット¥30,000、IEEEユニット¥30,000、トラクタユニット¥10,000、第2水準漢字ROM¥20,000

先駆のターミナルフロッピーTF-20の後継機種TF-10。●TF-20の高い信頼性と性能をさらにアップして低価格を実現しました。●PCシリーズ専用機(PC8001　、8001mkII、8801、9801)FMシリーズ専用機(FM7、8)MZ・FPシリーズ専用機(MZ-80B、2000、80K/C、1200、FP-1000、1100)等の各種専用機をラインナップ。●専用オプション基板を取り換えるだけで、簡単に他の専用機に変更できます。低価格で各種パソコンに対応が可能となりました。●本格的ビジネスユースからパーソナルユースまで用途に合わせてデュアルタイプ、シングルタイプの選択がができるほかシングルタイプはオプションの増設ドライブを加えることでデュアルタイプとなるなど幅広いユーザーニーズに対応しています。●コンパクトで薄型、フレキシブルディスクの出し入れは、片手で"押すだけ"のワンタッチ操作。簡単にディスクのセット、リセットができます。

■記憶容量(両面倍密度記憶方式の場合)：640KByte(デュアルタイプ)320KByte(シングルタイプ)■1ドライブ当りのトラック数：80トラック(40×2)■1トラック当りのセクタ数：16セクタ■1セクタ当りの記憶容量：256Byte■電源：電源電圧AC100V±10%■電波周波数：49.5～60.5Hz■消費電力：40VA以下■外形寸法：120W×350D×165Hmm■重量：7.0kg(デュアルタイプ)5.85kg(シングルタイプ)■価格：PCシリーズ専用機¥129,800(デュアルタイプ)¥89,000(シングルタイプ)、FMシリーズ専用機¥125,800(デュアルタイプ)¥85,000(シングルタイプ)　MZ・FPシリーズ専用機¥122,800(デュアルタイプ)¥82,000(シングルタイプ)〈オプション〉PCシリーズ専用基板¥29,000、FMシリーズ専用基板¥25,000、MZ・FPシリーズ専用基板¥22,000、増設ドライブ¥45,000



好評発売中	プリンタ操縦法入門	EPSON FP-80 (RP-80・MP-80)活用マニュアル

¥2,400/AB版 224ページ/ラジオ技術社発行/東京電機大学助教授 脇英世編著

あなたのパソコンとリンクする、ペリフェラルはエプソン。

ターミナルプリンタ

〈RPシリーズ〉

RP-80 ●MPの技術が生んだ高性能の経済派プリンタ¥89,000

RP-80F/T ●MP-80F/TIIIのシステムアップとコストダウンを実現した高精度ビットイメージプリンタ¥104,000

RP-100 11月発売 ●RP-80の機能に加え、インターフェイスを充実し、汎用性を高めたビットイメージプリンタ¥134,000

〈FPシリーズ〉

FP-80 ●高速印字と多彩な文字バリエーションの強力インテリジェントターミナル¥149,800(PC-8001専用¥152,800/PC-8001mkII、8801、9801専用¥153,800)FP-80用漢字ユニット#8175¥40,000(JISコードで漢字プリンタとして動作。)

FP-80K ●多彩な機能を搭載した高品質漢字プリンタ¥189,800(PC-8001専用¥192,800/PC-8001mkII、8801専用¥193,800)

FP-100 ●トータルバランスに秀れた万能型スーパープリンタ¥189,800(PC-8001専用¥192,800/PC-8001mkII、8801、9801専用¥193,800)

DP-100 11月発売 ●多彩な機能を搭載したデイジーホイールプリンタ¥140,000(パラレル仕様)¥150,000(シリアル仕様)〈オプション〉トラクタフィーダー¥24,000、オートカットシートフィーダー¥50,000、キーボードユニット¥39,000

ターミナルフロッピー

TF-20 ●APPLEII専用機デュアルタイプ¥128,000・シングルタイプ¥83,000(シングルタイプはオプションの増設ドライブを加えることでデュアルタイプとなります。¥50,000)

パーソナルカプラ

CP-20 ●コンピュータネットワーク、データ通信を強力にサポート。¥49,800

エプソン株式会社
●本社／〒399-07 長野県塩尻市広丘原新田80 ☎0263-52-2552代

エプソン販売株式会社 ●本社／〒160東京都新宿区西新宿2-4-1新宿NSビル内私書箱6109号☎03-348-7121(大代)・大阪支店／〒530大阪市北区西天満4-14-3住友生命御堂筋ビル11階☎06-365-5071・名古屋営業所／〒460名古屋市中区新栄町2-13栄第一生命ビル9階☎052-962-7001・札幌営業所／〒060札幌市中央区北一条西2丁目札幌時計台ビル6階☎011-222-2821・仙台営業所／〒980仙台市本町1-12-12山万ビル6階☎0222-63-3691・長野営業所／〒390松本市深志2-2-9加納ビル2階☎0263-36-7251・広島営業所／〒730広島市中区幟町3-1第3山県ビル4階☎082-223-2811・福岡営業所／〒812福岡市博多区博多駅前3-4-8郵政互助会博多ビル4階☎092-471-0761●詳しい資料のご請求は、はがきに住所、氏名、年令、職業、製品名をお書きの上、もよりのエプソン販売までお申込みください。

ペリフェラル
資料請求券
OH!/FM

Oh!FM Vol. 7
昭和五十九年二月一日発行 第二巻第一号通巻七号 昭和五十八年六月七日国鉄首都特別扱承認雑誌第六八九五号

Oh!FM 第7号 1984

●特集/ディスク活用術

●マニピュレータの製作

●マシン語ゲームの作り方

●菅子の住所録

日本ソフトバンク